Delinea

Server Suite

Planning and Deployment Guide

Version: 2024.x

Publication Date: 7/26/2024

Server Suite Planning and Deployment Guide

Version: 2024.x, Publication Date: 7/26/2024

Table of Contents

Planning and Deployment Guide

Most large-scale deployments rely on a project team to design and articulate a project plan, and team members take on specific roles and responsibilities. Depending on your role and responsibilities, you may want to read portions of this guide selectively.

> Most of the information in this guide applies to all platforms. However, there are some deployment scenarios and tasks that are unique to Mac OS X computers. If you manage Mac OS X computers and users, refer to the *Administrator's Guide for Mac* for additional information.

The guide provides the following information:

- Planning Deployment for an Enterprise provides an overview of key concepts and the deployment lifecycle, including suggestions for who should participate in the planning process and factors to consider that will affect your deployment strategy.
- Architecture and Basic Operations describes the key components of the Server Suite software architecture and how the components work together to provide authentication and authorization services.
- Deployment Process Overview provides an overview of the steps involved in a deployment project and a preview of the tasks you can expect to complete.
- Planning organizational units and security groups discusses the Active Directory objects and organizational model that is recommended to ensure a separation of duties for UNIX administrators.
- Installing Authentication & Privilege Services provides step-by-step instructions for installing and configuring Server Suite software components on Windows computers.
- Installing Agents on Computers to be Managed describes the installation options available and provides instructions for installing Server Suite software components on UNIX and Linux computers.
- Planning to use Server Suite zones describes the importance of zones and how you can use classic and hierarchical zone for identity management, access control, and delegated administration.
- Preparing To Migrate Existing Users And Groups describes the steps to take to prepare for migrating existing users and groups, including collecting and analyzing existing profile information and creating the first zone.
- Migrating Existing Users To Hierarchical Zones describes how to import and migrate an existing user population into hierarchical zones and enable authentication using Active Directory and Server Suite software.
- Joining Computers to a Domain and Zone describes how to complete the initial migration by joining the Active Directory domain and a Server Suite zone.
- Provisioning New User and Group Profiles After Migration describes how to use the Zone Provisioning Agent and Active Directory groups to automate provisioning of new users and groups.
- Validating Operations After Deploying provides suggestions for formal testing and validation activities you can perform to move from a pilot deployment to a production environment.
- Defining Role-Based Access for Users and Computers describes the most common roles that organizations create to complete the initial deployment and how to configure the appropriate rights and assign the roles to appropriate groups.

- Migrating And Managing Service Accounts describes the strategies you can use if you want to migrate local service accounts to Active Directory to improve security for those accounts.
- Planning to Deploy in a Demilitarized Zone (DMZ) describes how to deploy Server Suite components to allow communication between a perimeter (DMZ) zone and an internal zone.
- Managing and Evolving Operations After Deployment describes management activity for operations staff and additional services you may want to implement after deployment as you evolve the Server Suite software solution.
- Templates and Sample Forms provides examples of common documents and notification messages that you can customize and use throughout the deployment process.
- Permissions Required for Administrative Tasks provides information about the specific Active Directory permissions required to perform administrative tasks on objects specific to Server Suite.

Planning Deployment for an Enterprise

This section provides a brief review of the information you should have to begin planning a successful enterprise deployment of Server Suite. It includes an overview of the deployment life cycle, roles and responsibilities to consider in assembling a deployment team, and the factors you should consider during the planning phase that will affect how you deploy Centrify software.

For an overview of Centrify software and an introduction to basic tasks, see the *Evaluation Guide for Linux and UNIX*. For a general introduction to identity, access, and configuration management or more detailed information about performing administrative tasks, see the *Administrator's Guide for Linux and UNIX*.

What You Should Know Before Planning a Deployment

Before you begin planning a full scale deployment of Centrify software, you should be familiar with key concepts, terminology, and components for Server Suite and Active Directory. You should also have information about your existing environment.

Before you continue planning the deployment, verify you have information about:

- How Active Directory is used to store user, group, and computer information in your organization and the Active Directory schema you currently have deployed.
- How you currently manage services and provision users for both Windows and nonWindows computers.
- How the Centrify Agent installed on a UNIX, Linux, or Mac OS X computer makes that computer part of an Active Directory domain.
- How zones enable you to manage user profiles, control access to computer and application resources, and delegate administrative tasks.

If you are not familiar with Centrify architecture and the components that make up the Server Suite, see Architecture and basic operations to be sure you understand the concepts, core components, and operations that enable Active Directory users to log on to UNIX, Linux, and Mac OS X computers. This guide assumes you also have access to the *Administrator's Guide for Linux and UNIX* and can refer to it, as needed, for additional details.

Why Planning a Deployment is Important

Because Centrify software becomes a critical part of your IT infrastructure once deployed, it is important that you plan and test your deployment strategy and validate the results you expect before placing Centrify components into a production environment.

After you deploy Centrify software in a production environment, the rights and roles you define will control whether users can log on and what they can do on specific computers if they are allowed to log on. Because preventing users from accessing critical resources or services can affect business operations, you should analyze the requirements of your environment as thoroughly as possible before moving from a pilot deployment into production.

The deployment process described in this guide is intended to help you to migrate existing users and groups to Active Directory with minimal disruption to end-user activity and ongoing business services. The recommendations presented are designed to give you flexibility and provide you with a framework for deploying that minimizes the effect of the deployment on the existing user population.

Planning is important regardless of whether you are deploying on Windows, UNIX, Linux, or Mac computers. However, some deployment steps can be skipped if you are only deploying on Windows computers or if you aren't migrating any local users or groups. For more information about deploying only on Windows computers, see the *Administrator's Guide for Windows*. For information that is specifically about deploying on Mac computers, see the *Administrator's Guide for Mac*.

What to Expect During Deployment

In most organizations, a deployment takes place in the following stages:

Evaluation

A primary senior analyst or small group installs the software in an isolated test environment. The main goal of this stage is to learn basic concepts, terminology, and operations and validate any specific functionality that is critical to the organization adopting the software. The lab environment also allows you to test the planned changes to system and user management processes without affecting user access. This proof-of-concept stage often takes place before the decision to purchase the software or with the decision to purchase a small number of licenses for extended testing.

Analysis and Design

During this stage, a planning team does deeper analysis into the goals and requirements of the organization, the current state of the environment, and the deployment and management options that best suit the organization. The main goal of this stage is to design how you will use zones, import user account information, and assign rights and roles through a combination of Active Directory groups and zone definitions. Most of the information in this guide is intended to help you make those decisions and validate them in a pilot deployment.

Pilot Deployment

The pilot deployment is intended to be more robust than the evaluation stage. The pilot deployment is typically 10 to 20 computers, often with a common administrative owner or administrative group. The main goal of this stage is to verify your analysis accurately described your environment and to uncover any gaps that might have been missed or special circumstances that require adjustment to the design planned for zones, user account information, or

rights and roles. You can include more than 20 computers in the pilot deployment, but limiting the number makes the initial migration of the user population more manageable while you become familiar with the process.

Testing and Validation

After deploying the software, most organizations perform at least some formal testing of specific scenarios to ensure the authentication and authorization rules they have defined operate as expected and users are not locked out of computers they need access to but are prevented from logging on where they don't have access rights. The main goal of this stage is to execute a test plan that exercises software operations in a number of different use cases.

Roll-Out Deployment

After sufficient testing and verification of the pilot deployment, the deployment team can use a software delivery method to install Centrify Agent packages on remote computers and join an Active Directory domain. Typically, the roll-out is done in phases, so that Centrify software is deployed on a set of computers in one subnet, IP range, or administrative domain, then later deployed on another set of computers on a different subnet, with a different IP range, or in a different administrative domain. The goal of this stage is to deploy in a controlled manner, so that any issues can be resolved before they affect additional users or computers.

Ongoing Management and Evolution

As your environment changes and evolves, it is likely that you will want to refine, customize, and extend your deployment and your authentication, authorization, computer, and user management policies. You may also develop or enhance scripts that automate provisioning and decommissioning of accounts, or update business processes to take advantage of additional functionality, such as integration with other tools to capture Centrify data or configuring database applications to use PAM-based authentication. The goal of this stage is continuous improvement to streamline business processes and operational efficiency.

Preparing a Deployment Team

In large organizations, the network architecture and Active Directory infrastructure is often highly complex and sophisticated. Adding UNIX, Linux, and Mac OS X computers and users to this infrastructure requires careful planning and is handled best with a clearly documented deployment plan. This guide is intended to help you develop such a plan and to suggest the issues you should consider in designing a deployment that suits your organization. For an example of what a deployment plan might look like, see Simplified environment analysis and zone design template.

Depending on the size of your organization, you might want to assemble a cross-functional deployment team to plan and implement a deployment strategy, set up and test a pilot deployment program, and refine, document, and roll-out operations across the organization. In addition, a deployment team might include project leads and IT staff members who will be responsible for maintaining and managing Server Suite and Active Directory on an ongoing basis after deployment or developers who will extend or integrate applications to work with Server Suite and Active Directory.

A typical deployment team might consist of members in the following roles:

Active Directory Enterprise or Domain Administrators

Know the structure and trust relationships of one or more Active Directory forests, including the topology of the Active Directory site and the roles of the domain controllers. These administrators may also be responsible for provisioning and decommissioning accounts or maintaining the tools for these business processes.

UNIX Administrators or Administrators with Specific Expertise

Manage access for all or specific groups of UNIX, Linux, or Mac OS X computers. These administrators may be responsible for specific resources, such as the servers that host mission-critical applications or a web farm, or have specific knowledge, such Oracle database administration or AIX administrative tools.

Security Administrators

Establish security policies and audit trails and define the procedures for securing computer resources and user account information. These administrators may also define the provisioning rules for the organization or have detailed knowledge of the existing provisioning process.

IT or Network Architects

Understand the overall layout of the organization's network, including internal connectivity and access to the Internet, firewalls, port usage, bandwidth and latency issues.

Application Developers

Write programs that require authentication and authorization services. Application developers might also include UNIX programmers who will be responsible for writing scripts to automate administrative tasks, such as creating zones or adding new users to a zone.

Functional Testers

Develop test cases for the user scenarios the deployment team wants to validate.

Centrify Administrative Operators

Use Access Manager and other consoles on Windows, UNIX command line programs, ADEdit library, or PowerShell scripts to manage users, groups, computers, or zones. These operators might be delegated administrators for specific zones after deployment or existing Active Directory administrators who add and remove users from groups or manage Active Directory containers.

Database Administrators

Install and manage database instances and control access to database records. If you are planning a deployment that includes auditing user activity, the deployment team should include at least one database administrator to plan for and create the databases that will store captured sessions and audit meta-data. A database administrator can also provide procedures and guidance for backing up, archiving, and removing historical data as appropriate for your organization's record retention policies.

Internal or External Auditors

Understand regulatory compliance requirements for the organization and industry. Auditors typically know the type of information they need and can define the reports that will satisfy their needs.

Assembling a cross-functional team with members who have expertise in working with Active Directory and Windows architecture and members who have expertise in managing UNIX, Linux, or Mac OS X servers and workstations is often a key component of a successful deployment.

Preparing Deployment Documentation

In addition to deploying the software, the deployment team should prepare materials that document the solutions they are deploying and the processes and procedures to assist others in migrating. The deployment documentation might include training materials for new users and test plans to verify a successful deployment that can be reused when updating the software after the initial deployment.

In general, members of the deployment team should focus on the following activities to prepare for a roll-out of Server Suite to a production environment:

- Document the configuration settings you plan to use and update the documentation as needed based on the pilot experience. For example, during the planning phase you might have drafted a plan for user and group filtering or access controls that in practice you find must be adjusted. The pilot deployment gives you the opportunity to implement your planned solution but change it, if needed.
- Document and prototype any deployment scripts that you intend to use and any processes or policy decisions that impact using those scripts. For example, you might want to automate the join process or how new users are added to a zone or modify existing scripts that provision users.
- Document issues that require troubleshooting during the pilot deployment and the resolution for each issue. You can collect this information as an organization-specific operations manual for IT staff.
- Prepare training materials for testers, operations personnel, and end-users based on the experience gained in the pilot deployment and tailored to your organization's specific needs and internal policies.
- Prepare test plans that sufficiently cover the types of scenarios that are specific to your organization's needs and internal policies. For example, if you plan to use group policies, your test plans should include scenarios for testing the group policies you plan to implement.
- Update planning documents, such as the zone structure or role definitions that you developed during the planning phase in response to the practical experience gained in the pilot deployment.
- Create checklists or instructions that are specific to your organization's deployment. For example, you may want to create a "site preparation checklist" that covers specific steps administrators should take before deploying, a "deployment checklist" that includes site-specific naming conventions and migration instructions, and a "handoff to operations checklist" to ensure a smooth hand-over to data center staff after deployment is complete.

Defining Goals for the Deployment

One of the first tasks of the deployment team should be to define the goals you want to achieve and the criteria you will use to measure whether you have met those goals. As part of this process, you should define:

- **The primary reason for deploying Centrify in your organization.** For example, if providing centralized directory service or a single point of account administration is your most important goal, you may make different deployment decisions than if auditing and restricting user access to specific computers is your primary goal. That is, you want to be sure the deployment addresses your most pressing concerns first.
- **Priorities for any additional goals you want to set for the deployment.** For example, you may want to transition to a rationalized namespace over time, but this may be a lower priority for your organization than moving from decentralized computer administration to delegated administration of the tasks users can perform on specific computers.
- **Any specific auditing requirements or security requirements that are unique to your organization or industry.** For example, the way you organize computers into groups may be determined by specific reports you need to produce.
- **Internal policies for how you update and distribute software.** For example, you should define how frequently you apply operating system patches and whether you automate software distribution.
- **Internal policies for how you assign UNIX attributes and Active Directory account information.** For example, you should identify how you have assigned UIDs, GIDs, and other UNIX-specific attributes and whether there are existing naming conventions for Active Directory users and groups.
- **Plans for who will manage UNIX profiles after deployment.** For example, you should identify the group or groups that will manage which UNIX users and computers and whether there will be separate UNIX and Active Directory administrators with shared responsibilities or a clearly defined division of responsibilities. In most cases, Centrify recommends a separation of duties model that enables UNIX administrators to manage zones and Active Directory administrators to manage user objects and group membership.

Architecture and Basic Operations

This section provides an overview of the Server Suite architecture and the components for Windows and non-Windows computers. It also describes the basic flow of operation when users log in or access applications, and what happens when an Active Directory domain controller goes down.

The information in this section is not required for planning a deployment. It is intended as background information that can help you understand the authentication and authorization process in some detail. If you want to proceed directly to planning the deployment, you can skip this section.

Server Suite Platform-Specific Components

Server Suite provide an integration layer between Active Directory in a Windows environment and computers running other operating systems or application environments. Because of this, Server Suite includes components for managing Active Directory-based objects in the Windows environment and agents that run on each server or workstation to be integrated into Active Directory.

Server Suite Components for Windows

On Windows, Server Suite includes management consoles and services to simplify the management and integration of Linux and UNIX computers and users into Active Directory.

The key components for Windows that you use in deployment are:

- Access Manager console
- Zone Provisioning Agent configuration panel and Windows service

There are several additional Windows components available for you to use, depending on the version of Server Suite software you install and the requirements of your environment. For example, Server Suite offers extensions for working with NIS maps and Active Directory group policies, as well as components to support a multi-tier architecture for auditing activity in user sessions and the Network Information Service to support agentless authentication service.

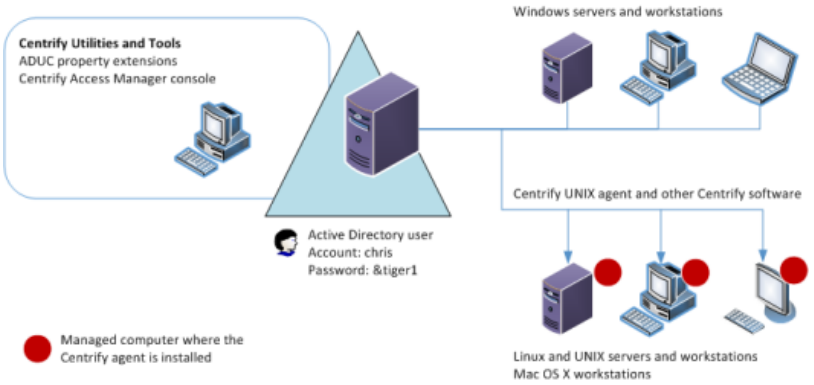
Components Installed on Managed Computers

On non-Windows computers, Server Suite software consists of the core Server Suite Agent (adclient), related libraries, and optional tools. The Server Suite Agent enables the local host computer—most commonly a Linux or UNIX computer—to join an Active Directory domain.

After the agent is deployed on a server or workstation, that computer is considered a **managed computer** and it can join any Active Directory domain you choose.

When a Server Suite-managed computer joins an Active Directory domain, it essentially becomes an Active Directory client and relies on Active Directory to provide authentication, authorization, policy management, and directory services. The interaction between the agent on the local computer and Active Directory is similar to the interaction between a Windows workstation and its Active Directory domain controller, including failover to a backup domain controller if the managed computer is unable to connect to its primary domain controller.

The following figure provides a simplified view of the integration between Windows and non-Windows computers through Server Suite software.



To use Microsoft Active Directory to centrally manage access across different platforms, you need to do the following:

- Prepare the Active Directory environment by installing the Access Manager console on at least one Windows computer and using the Setup Wizard to update the Active Directory forest.
- Ensure each UNIX, Linux, or Mac OS X computer can communicate with an appropriate Active Directory domain controller through DNS.
- Install the agent (adclient) on the UNIX, Linux, or Mac OS X computers that will be joining an Active Directory domain.

- Run the join command and specify the Active Directory domain on each UNIX, Linux, or Mac OS X computers that needs to join an Active Directory domain.
- Use Active Directory Users and Computers or Access Manager to authorize access to the UNIX, Linux, and Mac OS X computers for specific users and groups.

The next sections provide a more detailed discussion of the Server Suite architecture and a summary of what happens when a user logs on to a UNIX computer that has joined the Active Directory domain.

Storing Server Suite Properties in Active Directory

The Active Directory schema defines the object classes that can be stored in Active Directory, and the attributes that each object class must have, plus any additional attributes the object can have, and the object class that can be its parent. Schema definitions are also stored as objects in Active Directory. To store UNIX-specific attributes within the Active Directory schema, the schema must be able to include the properties that are associated with a UNIX user or group. For example, for a UNIX user, the schema needs to accommodate the following information fields:

- UNIX user name
- Password hash (optional)
- Numeric user identifier (UID)
- Primary group identifier (GID)
- General information (GECOS)
- Home directory
- Default shell

Some of these information fields are similar to standard user class attributes in Active Directory. For example, the Active Directory Display Name (displayName) attribute typically stores a user's full name—the same information typically stored in the GECOS field in an /etc/passwd file on a UNIX computer, so the displayName is used to define the contents of the GECOS field in a user's UNIX profile. Depending on the Active Directory schema you have installed, some of the information fields required for logging on to UNIX computers might not have an equivalent Active Directory attribute.

If you are using the default Active Directory schema, Server Suite stores UNIX-specific attributes in an Active Directory class under its own parent container for zones. Server Suite then organizes the information about individual UNIX computers, users, and groups by zone.

If your organization has already deployed a Microsoft-supported set of UNIX schema extensions, such as those defined in the Windows **Services for UNIX** (SFU) schema extension, you can store UNIX attributes in the fields defined by that schema as an alternative to using the zones container.

If you have deployed the **RFC 2307-compliant** Active Directory schema, you can store UNIX attributes in the fields defined by that schema and organized into RFC 2307-compliant zones.

After you have installed Server Suite components on a Windows computer, the first time you open the Access Manager administrative console, a Setup Wizard updates the Active Directory forest to include the Server Suite properties for UNIX attributes. You can then use Access Manager, the Active Directory Users and Computers MMC snap-in, ADEdit commands, or PowerShell scripts to view and modify the UNIX properties for any user, group, or computer.

For RFC 2307-compliant zones, the group name and UNIX name are stored in the same CN attribute. Therefore, if you change a group's name with its Active Directory Users and Computers' property page, the UNIX name is changed in Access Manager as well.

Using Access Manager

Access Manager is the primary user interface for managing all of the Server Suite-specific information stored in Active Directory. With Access Manager, you can:

- Manage access to all of your UNIX, Linux, and Mac OS X computers.
- Set and modify user and group properties for all of your UNIX, Linux, and Mac OS X users and groups.
- Create and manage zones and zone properties to simplify the process of giving users access to specific computers and migrating UNIX user accounts to Active Directory.
- Add Active Directory users and groups to zones.
- Import user and group information from local password and groups files or from NIS and NIS+ servers and domains.
- Import and maintain network information from NIS maps such as netgroup, auto.master, and automount or create custom NIS maps.
- Define and assign rights and roles that authorize or restrict access to specific computers and operations on managed computers.

You can also add other snap-ins to Access Manager or add Access Manager to another Microsoft management console snap-in. For example, you can add the Active Directory Sites and Services and Active Directory Domains and Trusts snap-ins to Access Manager to consolidate management activity.

Allowing and Blocking Domains for Access Manager

You can configure Access Manager so that it can connect to trusted domains by setting the following registry key with a list of trusted domains and/or forests. The type of key is REG_MULTI_SZ:

HKLM\SOFTWARE\Centrify\CIMS\AllowedTrusts

Configuring a list of domains this way can be particularly useful and faster when you have a large amount of domains. Enter each domain as a separate line in the Registry Editor window.

For example, to specify a single domain:

acme.com

For example, to specify multiple domains:

acme.com
foo.com

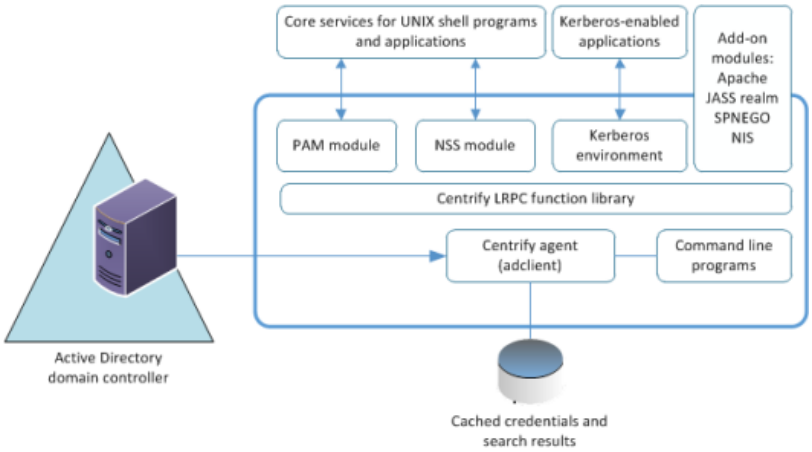
To block access to domains, you use the IgnoreTrusts key: HKLM\SOFTWARE\Centrify\CIMS\IgnoreTrusts.

Core Agent Components and Services

The Server Suite Agent makes a UNIX, Linux, or Mac OS X computer look and behave like a Windows computer to Active Directory. Once installed, the agent performs the following key tasks:

- Joins UNIX, Linux, or Mac OS X computers to an Active Directory domain.
- Communicates with Active Directory to authenticate users logging on to the UNIX, Linux, or Mac OS X computer, and caches credentials for offline access.
- Enforces Active Directory authentication and password policies.
- Extends Active Directory group policies to manage the configuration of UNIX users and computers.
- Provides a Kerberos environment so that existing Kerberos applications work transparently with Active Directory.

Individual agents are platform-specific, but provide an integrated a set of services to extend Active Directory authentication, authorization, and directory service to managed computers. The following figure provides a closer look at the services provided through the Server Suite Agent:



As this figure suggests, the agent typically includes the following core components:

- The core component of the agent is the adclient process that handles all of the direct communication with Active Directory. The agent contacts Active Directory when there are requests for authentication, authorization, directory assistance, or policy updates, and then passes valid credentials or other requested information along to the programs or applications that need this information.
- The **Centrify Pluggable Authentication Module**, pam_centrifydc, enables any PAM-enabled program, such as ftpd, telnetd, login, and sshd, to authenticate using Active Directory.

> For AIX and Mac OS X, the implementation is slightly different. For example, the agent for AIX can use PAM interfaces if you have configured the computer to use PAM modules or the interfaces in the Loadable Authentication Module (LAM) to handle behavior that on other platforms is done through PAM or NSS. Similarly, the agent for Mac OS X uses native interfaces where appropriate to provide services from Active Directory to the local computer.

- The **Centrify NSS** module is added to nsswitch.conf so that system look-up requests use the agent to look up and validate information using Active Directory through LDAP.

- The **ADEdit Tcl application and procedure library** and **individual UNIX command line programs** enable you to perform common administrative tasks, such as join and leave the Active Directory domain or change user passwords for Active Directory accounts interactively or within scripts to automate tasks.
- The **Server Suite-managed Kerberosenvironment** generates a Kerberos configuration file (etc/krb5.conf) and a default key table (krb5.keytab) file to enable your Kerberos-enabled applications to authenticate through Active Directory. These files are maintained by the agent and are updated to reflect any changes in the Active Directory forest configuration.
- The **Server Suite local cache** stores user credentials and other information for offline access and network efficiency.

In addition to these core components, the agent can also be extended with the additional software packages, including modified versions of programs such as Kerberos command line tools, OpenSSH, OpenLDAP, and PuTTY utilities. Server Suite-enabled versions of these programs allow you to use Active Directory accounts and Kerberos credentials for authentication, authorization, and policy enforcement services. Server Suite also provides authentication modules that enable you to configure single sign-on for web and database applications, and specialized extensions such as the adnisd Network Information Service, which enables you to publish information stored in Active Directory to NIS clients.

Key Operations Handled by the Adclient Process

The most important element in the agent is the adclient process. The adclient process runs as a single trusted service. This process is automatically added as a boot service and is started whenever you reboot a managed computer. The adclient process handles all of the direct communication with Active Directory and manages all of the operations provided through the other services.

The adclient process performs the following key tasks on managed computers:

- Locates the appropriate domain controllers for the local computer based on the Active Directory forest and site topology published by the Windows DNS server. If a domain controller becomes unavailable, the adclient process automatically locates the next available domain controller to ensure uninterrupted service.
- Provides Active Directory with credentials for the local computer account to verify the computer is a valid member of the domain.
- Delivers and stores user credentials so that users can be authenticated by Active Directory and, once authenticated successfully, can sign on even if the computer is disconnected from the network for mobile access or if Active Directory is unavailable.
- Caches query responses and other information, including positive and negative search results, to reduce network traffic and the number of connections to Active Directory and to ensure users can work uninterrupted and start new application sessions using their existing login credentials. All communication with Active Directory is encrypted to ensure security, and you can manage the cache through configuration parameters or group policy.
- Creates and maintains the Kerberos configuration and service ticket files to allow existing Kerberos-enabled applications to work with Active Directory without any manual configuration.
- Synchronizes the local computer's time with the clock maintained by Active Directory to ensure the timestamp on Kerberos tickets issued by the KDC are within a valid range.

- Resets the password for the local computer account in Active Directory at a regular interval to maintain security for the account's credentials.
- Provides all the authentication, authorization, and directory look-up services retrieved from Active Directory to the other Server Suite Agent services, such as the PAM service or the Apache authentication module.

How PAM Applications Work with Server Suite

Pluggable Authentication Modules (PAM) are a common mechanism for configuring authentication and authorization used by many UNIX programs and applications. If a program or application uses PAM for authentication and authorization, the rules for authenticating the user are configured in either the PAM configuration file, /etc/pam.conf or in application-specific files in the /etc/pam.d directory.

The Server Suite Agent for *NIX includes its own Pluggable Authentication Module (pam_centrifydc) that enables any application that uses PAM, such as ftpd, telnetd, login, and Apache, to authenticate users through Active Directory. When you join a domain, the pam_centrifydc module is automatically placed first in the PAM stack in systemauth, so that it takes precedence over other authentication modules.

The pam_centrifydc module is configured to work with adclient to provide a number of services, such as checking for password expiration, filtering for users and groups, and creating the local home directory and default user profile files for new users. The services provided through the pam_centrifydc module can be customized locally on a computer, modified through Active Directory group policy, or configured through a combination of local and Active Directory settings.

Working in conjunction with the adclient process, the pam_centrifydc module provides the following services for PAM-enabled programs and applications:

- Requests the PAM-enabled application to prompt for a password when appropriate and verifies whether the application-provided user name and password are valid in Active Directory.
- Checks whether the user's password has expired in Active Directory. If the password has expired, the pam_centrifydc module prompts the user to change the password, and forwards the new password to the adclient process, which communicates the change to Active Directory.
- Queries the adclient process to determine whether any access control policies are applied. For example, the pam_centrifydc module uses the information in the centrifydc.conf file to determine whether a local user attempting to log on is mapped to an Active Directory account, whether specific users or groups have been granted or denied permission to log on to the local computer, or whether Active Directory authentication should be ignored for a specific user or group.
- Creates the local home directory and default user profile files for new users. The pam_centrifydc module uses skeleton files to set up the user environment when new Active Directory users log on to a managed computer for the first time.

Most of these tasks are performed during a user login session as a series of requests and replies from the pam_centrifydc module to Active Directory through the adclient process for those programs and applications that are configured to use PAM. Because PAM is the most common authentication service used by UNIX programs and applications, the pam_centrifydc module is the most commonly used for a typical log-on session. For a more detailed description of a typical log-on process, see What happens during the typical log-on process.

The order in which identity stores are listed in the nsswitch.conf file does not influence authentication. Authentication and authorization services are provided by Active Directory through the Server Suite Agent for *NIX and its PAM component, and by default, Active Directory is always tried before any other sources. The order in which sources are checked is controlled through the PAM configuration settings, for example, the lines defined in the pam.conf file. In general, you should not modify the PAM configuration because making changes to these settings can compromise security or produce unexpected and undesirable results.

How NSS Configuration Works with Server Suite

The Name Service Switch (NSS) provides a mechanism for identifying sources of network information a computer should use, such as local password and group files, NIS maps, NIS+ tables, LDAP, and DNS, and the order in which these sources should be consulted when looking up users, groups, host names, and other information.

When you join a domain, the NSS configuration file, nsswitch.conf, is automatically updated to use the Server Suite Agent's NSS module first. Using the adclient process and the service library, the Server Suite NSS module accesses network information that's stored in Active Directory through LDAP.

When a UNIX program or application needs to look up information, it checks the nsswitch.conf file and is directed to use the nss_centrifydc module. The nss_centrifydc module directs the request to Active Directory through the adclient process. The adclient process provides the information retrieved from Active Directory, then caches the information locally to ensure faster performance, reduce network traffic, and allow for disconnected operation.

The order in which identity stores are listed in the nsswitch.conf file does not influence authentication. Authentication and authorization services are provided by Active Directory through the Server Suite Agent and its PAM service, so Active Directory is always tried before any other sources, regardless of what you have specified in the nsswitch.conf file. Instead, the nsswitch.conf file determines the sources to use in responding to NSS queries such as getpwnam. In general, you should not modify this file because modifying the file can compromise security and complicate auditing activity. In addition, you should not specify ldap as a source in any nsswitch.conf file where you have installed the Server Suite Agent. Specifying ldap in the nsswitch.conf file can cause the system to crash.

How the Server Suite Agent Manages Kerberos Files

Kerberos is a network authentication protocol for client/server applications that uses encrypted tickets passed through a central Key Distribution Center to verify the identity of a user or service requesting access. Because Kerberos is an industry standard and a secure network authentication mechanism, you may already have UNIX programs and services that are configured to use it. To allow those existing Kerberized applications to work with Active Directory without manual configuration, the adclient process automatically creates and maintains the Kerberos configuration file, krb5.conf, and the krb5.keytab service ticket file to point Kerberos-enabled services and applications to the Key Distribution Center (KDC) in Active Directory when you join a domain.

The configuration file is initially created using information collected by probing DNS and Active Directory with the default domain set to the domain that the computer has joined. Whenever a logon or ticket validation is performed with a domain that is not in the configuration file, the configuration file is updated so that it includes the new domain. Although the adclient process can automatically update the file as needed, it does not destroy existing configuration entries that you may have added by hand. Because of this, Server Suite Agents work seamlessly with existing Kerberos-enabled applications.

The Authentication Service supports users defined in a Kerberos realm as long as the Kerberos domains or realms are resolvable by DNS. Kerberos realm names are case sensitive, so be careful to check that the realm spelling and capitalization is correct. (Ref: CS-21846a)

What Happens During the Typical Log-on Process

The core Server Suite Agent for *NIX components work together to identify and authenticate the user any time a user logs on to a computer using any UNIX command that requires the user to enter credentials. The following steps summarize the interaction to help you understand the process for a typical log on request. The process is similar, though not identical, for UNIX commands that need to get information about the current user or group.

The following steps focus on the operation of the agent rather than the interaction between the agent and Active Directory. In addition, these steps are intended to provide a general understanding of the operations performed through the agent and do not provide a detailed analysis of a typical log on session.

When a user starts the UNIX computer, the following takes place:

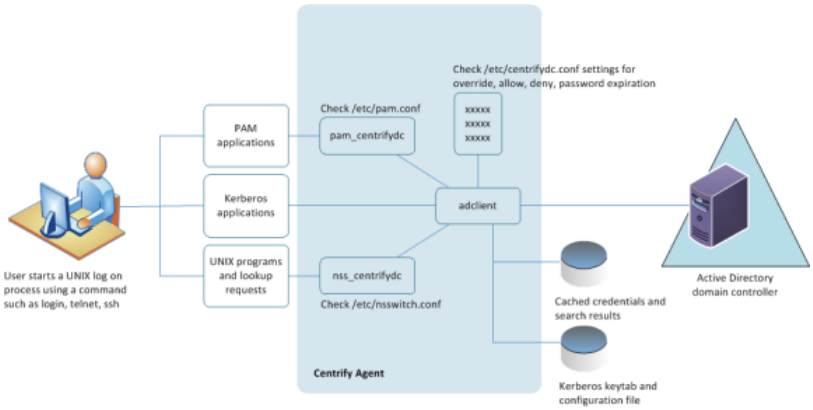
1. A login process starts and prompts the user to supply a user name.
2. The user responds by entering a valid local or Active Directory user name.
3. The login process, which is a PAM-enabled program, then reads the PAM configuration file, /etc/pam.conf, and determines that it should use the Server Suite PAM service, pam_centrifydc, for identification.
4. The login process passes the login request and the user name to the Server Suite PAM service for processing.
5. The pam_centrifydc service checks the pam.allow.override parameter in the agent configuration file to see if the user name entered is an account that should be authenticated locally.
   - If the user should be authenticated locally, the pam_centrifydc service passes the login request to the next PAM module specified in the PAM configuration file, for example, to the local configuration file /etc/passwd.
   - If the user is not listed as an override account, the pam_centrifydc service continues with the login request and checks to see if the adclient process is running, then passes the login request and user name to adclient.
6. The adclient process connects to Active Directory and queries the Active Directory domain controller to determine whether the user name included in the request is a user who has access to computers in the current computer's zone.
   - If the adclient process is unable to connect to Active Directory, it queries the local cache to determine whether the user name has been successfully authenticated before.
   - If the user account does not have access to computers in the current zone or can't be found in Active Directory or the local cache, the adclient process checks the Server Suite Agent configuration file to see if the user name is mapped to a different Active Directory user account with the adclient.mapuser.username parameter.

- If the user name is mapped to another Active Directory account in the configuration file, the adclient process queries the Active Directory domain controller or local cache to determine whether the mapped user name has access to computers in the current computer's zone.

7. If the user has a UNIX profile for the current zone, the adclient process receives the zone-specific information for the user, such as the user's UID, the user's local UNIX name, the user's global Active Directory user name, the groups of which the user is a member, the user's home directory, and the user's default shell.
8. The adclient process checks for NSS override settings (nss.group.override and nss.user.override) to determine whether there are any changes to the user profile or additional restrictions that should override the profile retrieved or prevent the user from logging on.
9. The adclient process queries through the nss_centrifydc service to determine whether there's another user currently logged in with same UID.
   - If there is a potential conflict between local user account and the UNIX profile for an Active Directory account, the adclient process notifies the pam_centrifydc service of the potential conflict.
   - The pam_centrifydc service checks the Server Suite Agent configuration file to determine to issue a warning, ignore the conflict, or prevent the user from logging on.
   - If the login continues, the pam_centrifydc service asks the login process for a password.
10. The login process prompts the user to provide a password and returns the password entered to the pam_centrifydc service.
11. The pam_centrifydc service checks the pam.allow.users and pam.deny.users parameters in the agent configuration file to see if any user filtering has been specified to allow or deny access to specific user accounts. If any user filtering has been specified, the current user is either allowed to continue with the login process or denied access.
12. The pam_centrifydc service checks the pam.allow.groups and pam.deny.groups parameters in the agent configuration file to see if any group filtering has been specified to allow or deny access to members of specific groups. If any group filtering has been specified, the current user is either allowed to continue with the login process or denied access based on group membership.
13. If the current user account is not prevented from logging on by user or group filtering, the pam_centrifydc service queries the adclient process to see if the user is authorized to log on.
14. The adclient process queries the Active Directory domain controller through Kerberos to determine whether the user is authorized to log on to the current computer at the current time.
15. The adclient process receives the results of its authorization request from Active Directory and passes the reply to the pam_centrifydc service.
16. The pam_centrifydc service does one of the following depending on the content of the authorization reply:
    - If the user is not authorized to use the current computer or to log in at the current time, the pam_centrifydc service denies the user's request to log on through the UNIX login process.
    - If the user's password has expired, the pam_centrifydc service sends a request through the UNIX login process asking the user to change the password. After the user supplies the password, the login process completes successfully.

- If the user's password is about to expire, the pam_centrifydc service notifies the user of impending expiration through the login process.
- If the user is authorized to log on and has a current password, the login process completes successfully. If this is the first time the user has logged on to the computer through the agent, the pam_centrifydc service creates a new home directory on the computer in the location specified in the agent configuration file by the parameter pam.homeskel.dir.

The following figure provides a simplified view of a typical log-on process when using the Server Suite Agent for *NIX.



How Failover and Disconnected Access Work

The Server Suite Agent caches data from Active Directory so that users can log on and perform tasks even if the network or Active Directory server is unavailable, whether because of unexpected connectivity problems, scheduled maintenance, or offline operation of a portable computer. There are several configuration parameters that manage how the agent determines its connectivity to Active Directory, the domain controllers it should attempt to connect to, and the operation of the agent if it is unable to connect to any domain controller.

In most cases, you can set the values for the configuration parameters that control failover and disconnected operation by enabling Server Suite group policies for a site, domain, or organizational unit. Alternatively, you can set these parameters by editing the /etc/centrifydc/centrifydc.conf configuration file on individual computers.

For an overview of how the agent determines the connection status and locates a domain controller to use, see the following topics:

- Establishing a connection to DNS
- Connecting to the closest domain controller
- Restricting the domain controllers contacted
- Switching to disconnected mode
- Responding to DNS configuration changes
- Connecting to trusted forests and domains

Establishing a Connection to DNS

With each request to Active Directory, the Server Suite Agent first determines its connection status based on upon the availability of a Domain Name Service domain controller. If a DNS request for a host name takes longer than the number of seconds specified by the adclient.dns.response.maxtime parameter, the agent assumes DNS is down and switches to disconnected mode.

While running in the disconnected mode, the agent does not attempt any more synchronous network communications. Instead, it runs a background thread every 30 seconds to determine when DNS becomes available. The default value for the adclient.dns.response.maxtime is 10 seconds, but this value can be changed by group policy or by editing the /etc/centrifydc/centrifydc.conf file.

If the network is disconnected for a short period of time, but during that time no data is needed from Active Directory, the agent does not switch into disconnected mode. The status only changes if a connection attempt to DNS or to Active Directory through LDAP fails.

Connecting to the Closest Domain Controller

If the initial DNS request for a host name is successful, the Server Suite Agent attempts to connect to the appropriate domain controller and global catalog for its joined domain using the **site information** found in DNS.

Site information is configured using Active Directory Sites and Services and is defined by subnet. Using the site information, the agent queries DNS for a list of the domain controllers in its site and attempts to connect to the nearest domain controller. It will continue trying to connect to each of the domain controllers in its site based on proximity until it finds a server available. If the agent is unable to connect to any of the domain controllers in its site or if no site information is available, the agent tries to connect to any remaining domain controllers listed in DNS.

Because connection status is determined by an attempt to bind to the Active Directory domain controller using an LDAP call, the adclient.ldap.socket.timeout parameter determines the maximum number of seconds the Server Suite Agent will wait for a socket connection timeout while binding to the LDAP server. The default value is 5 seconds.

Restricting the Domain Controllers Contacted

If you have a large Active Directory infrastructure or some unreliable subnets, you might want to restrict the domain controllers the agent should attempt to connect to if its primary domain controller becomes unavailable. You can limit the list of domain controllers the agent should attempt to connect to by setting the following property in the centrifydc.conf file:

dns.dc.domain_name: hostname [hostname] ...

where the domain_name is the Active Directory domain name and the hostname is a fully-qualified host name that can be resolved using DNS or the /etc/hosts file.

You can also limit the list of global catalog domain controllers the agent should attempt to connect to by setting the following property in the centrifydc.conf file:

dns.dc.forest_name: hostname [hostname] ...

where the forest_name is the forest root domain and the hostname is a fully-qualified host name that can be resolved using DNS or the /etc/hosts file.

Alternatively, you can use the adclient.server.try.max parameter or Maximum Server Connection Attempts group policy to limit the number of domain controllers the agent will attempt to connect to before switching to disconnected mode, eliminating the need to explicitly list the domain controllers using the dns.dc.domain_name and dns.gc.forest_name parameters. For example, to have the agent try a maximum of three domain controllers, you can set the following property in the centrifydc.conf file:

adclient.server.try.max: 3

Because global catalog and domain controller connections are handled independently, Server Suite Agent for *NIX can still provide authentication services if the global catalog domain controller is disconnected, as long as another domain controller is available.

Switching to Disconnected Mode

After a connection to a domain controller is established, each subsequent request for information from Active Directory checks the connection status. If a request is made to Active Directory and a response is not received within the number of seconds specified by the adclient.ldap.timeout parameter, that request is retried once. For the second request, the agent will wait up to twice as long for a response. If the second request is not answered within that amount of time, the connection to that specific domain controller is considered disconnected. Once a connection to a specific domain controller is in disconnected mode, a background thread will attempt to reconnect to that domain approximately every 30 seconds. By default, the agent waits 7 seconds for a response to the first request. If the request isn't answered, it retries the request and waits up to another 14 seconds for a response before switching to disconnected mode.

The adclient.ldap.timeout parameter specifies the maximum number of seconds to wait for Active Directory fetch, update, and delete requests to improve the response time when an initial connection attempt fails. A separate parameter, adclient.ldap.timeout.search, specifies the maximum time to wait for search requests. If the search timeout value is not specified, the default is double the adclient.ldap.timeout value. By default, therefore, the agent waits a maximum of 14 seconds for search requests.

The values for these parameters can be adjusted for high load or latency networks by configuring group policies or by editing the /etc/centrifydc/centrifydc.conf file.

Responding to DNS Configuration Changes

The DNS information collected when the agent starts and connects to a domain controller is not cached, and idle connections to Active Directory are dropped after 5 minutes by default. If you make changes in the DNS configuration, those changes are detected the next time the agent needs to reconnect, either because an idle connection has been dropped, or the currently connected domain controller suddenly becoming unavailable.

Connecting to Trusted Forests and Domains

If the Server Suite Agent establishes a successful connection to the joined domain, it also generates or updates the /etc/krb5.conf file using the domain trust information from the global catalog, and attempts to connect to the trusted domains or to external forests to find all of the domains that are trusted.

Depending on the trust relationships you have defined, network topology, or firewall requirements, querying external trusted forests can have a significant, negative impact on network performance. You can control whether trusted domains and external forests are queried to establish transitive trusts and cross-forest authentication with the adclient.ldap.trust.enabled parameter. Setting the adclient.ldap.trust.enabled parameter to true indicates that

you want the Server Suite Agent to query trusted domains and forests. Setting this parameter to false disables this feature so that the agent does not connect to any external forests or trusted domains.

If you set the adclient.ldap.trust.enabled parameter to true, you can control the maximum number of seconds to wait when searching for trust information in external forests and other trusted domains with the adclient.ldap.trust.timeout parameter. By default, the agent waits 10 seconds. The search operation is not retried if the request times out, but the request is regenerated approximately once an hour.

If your trusted domains and forests are widely distributed, have slow or unreliable network connections, or are protected by firewalls, you might want to increase the value for this parameter to allow time for the Server Suite Agent to collect information from external domains and forests.

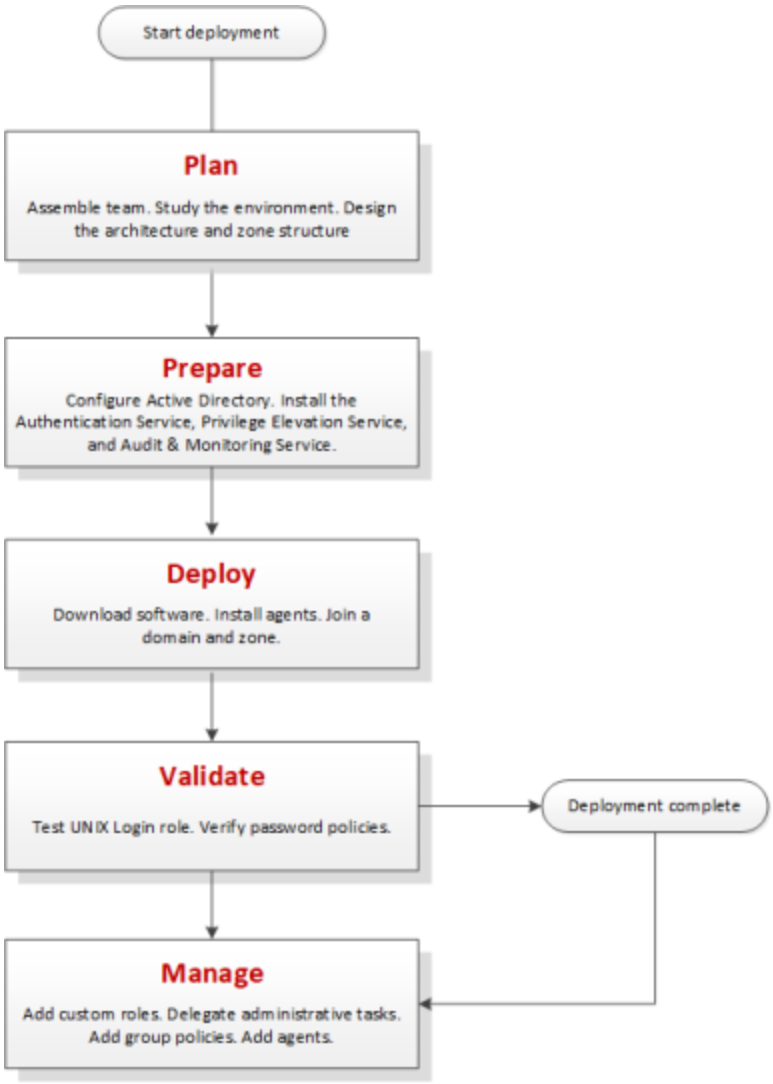
Deployment Process Overview

This chapter summarizes what's involved in deploying Centrify software. It includes simplified diagrams that highlight the steps involved and describes the tasks that are done only once, the tasks that are repeated to complete a deployment, and the tasks that may be part of the deployment project or ongoing administration after deployment.

The individual diagrams provide additional details about what's involved in each phase or the decisions you will need to make, such as who should be part of the deployment team, where to install the software, and who has permission to do what.

What's Involved in a Typical Deployment Project

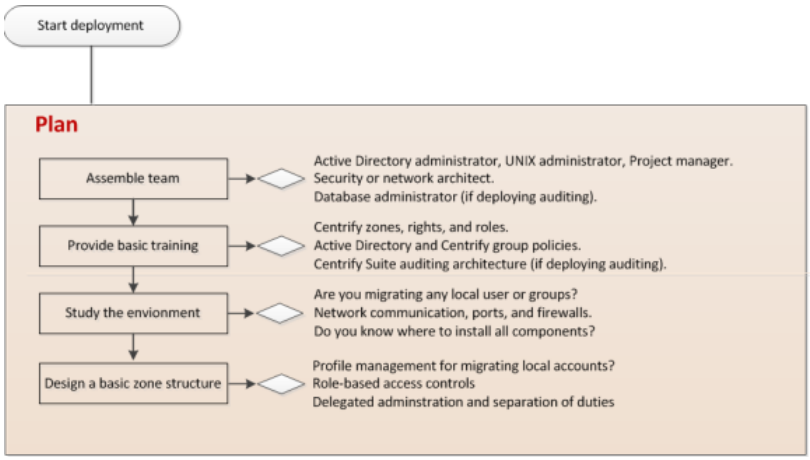
The following illustration provides a visual summary of the overall deployment process and highlights a few keys to a successful deployment.



The next sections provide additional details about each of these phases.

Plan

During the first phase of the deployment, you should collect and analyze details about your organization's requirements and goals. You can then make preliminary decisions about sizing, network communication, and what your zone structure should look like.



Here are the key steps involved:

- Assemble a deployment team with Active Directory and UNIX expertise.

  The team might also include specialists, such as database administrators, network architects, or application owners. For more information about assembling a deployment team, see Preparing a deployment team.
- Provide basic training so that members of the deployment team are familiar with Centrify concepts and terminology and know where to go for more information.
- Analyze the existing environment to determine your goals and requirements and identify target computers on which you plan to install Centrify components.

  This step is essential for designing the zone structure if you are migrating any local accounts or legacy profiles. It is also critical if you are deploying the auditing infrastructure. For more information about the questions to answer and factors that affect deployment, see Defining goals for the deployment.
- Design a basic zone structure that suits your organization.

  The zone structure depends primarily on how you want to use zones. For more information about deciding how to use zones, see Why use zones?.
- Identify a target set of computers for deployment and check that required ports are open.

Default Ports for Network Traffic and Communication

To help you plan for network traffic, the following ports are used in the initial set of network transactions when a user logs on and the agent connects to Active Directory:

- Directory Service - Global Catalog lookup request on port 3268.
- Authentication Services - LDAP sealed request on port 389.
- Kerberos – Ticket Granting Ticket (TGT) request on port 88.
- Network Time Protocol (NTP) Server – Time synchronized for Kerberos on port 123.
- Domain Name Service (DNS) – Host (A), Pointer (PTR), Service Location (SRV) records on port 53.

Depending on the specific components you deploy and operations performed, you might need to open additional ports. The following table summarizes the ports used for different editions of Centrify software.

This port	Is used for	Where it is required
389	Encrypted TCP/UDP communication	Centrify authentication service and privilege elevation service for Active Directory authentication and client LDAP service.
3268	Encrypted TCP communication	Centrify authentication service and privilege elevation service for Active Directory authentication and LDAP global catalog updates.
88	Encrypted UDP communication	Centrify authentication service and privilege elevation service for Kerberos ticket validation and authentication for agents and Centrify PuTTY.
464	Encrypted TCP/UDP communication for Kerberos password changes	Centrify authentication service and privilege elevation service for Kerberos ticket validation and authentication for agents, Centrify PuTTY, adpasswd, and passwd.
53	TCP/UDP communication	Centrify authentication service and privilege elevation service for clients using the Active Directory DNS server role for DNS lookup requests.
445	Encrypted TCP/UDP communication for delivery of group policies	Centrify authentication service and privilege elevation service for adclient and adgpupdate using Samba (SMB) and Windows file sharing to download and update group policies, if applicable.
123	UDP communication for simple network time protocol (NTP)	Centrify authentication service and privilege elevation service to keep time synchronized between clients and Active Directory for Kerberos ticketing.
22	Encrypted TCP communication for OpenSSH connections	Centrify authentication service and privilege elevation service to support secure shell connections on remote clients.
23	TCP communication for Telnet connections	Centrify authentication service and privilege elevation service to support telnet connections on remote clients if you cannot use secure shell (ssh). By default, telnet connections are not allowed because passwords are transferred over the network as plain text.
none	ICMP (ping) connections	Centrify authentication service and privilege elevation service to determine whether if a remote computer is reachable.
1433	Encrypted TCP communication for the collector connection to Microsoft SQL Server	Centrify authentication service, privilege elevation service, and audit and monitoring service to enable the collector service to send audited activity to the database.

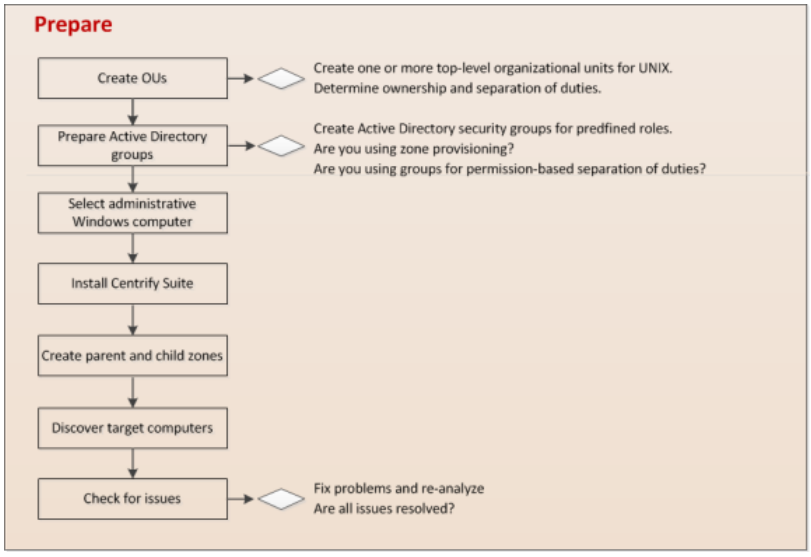
5063	Encrypted TCP/RPC communication for the agent connection to collectors	Centrify authentication service, privilege elevation service, and audit and monitoring service to enable the auditing service to record user activity on an audited computer.
443	Cloud proxy server to Centrify cloud service	Centrify for mobile device management.

Network Connections and Database Management for Auditing

If you are planning a deployment with audit and monitoring service installed together with identity and privilege management, you must plan for reliable, high-speed network connections between components that collect and transfer audit data and how network traffic will be affected. You must also plan how you will create and manage the databases that store and retrieve audit data, your data archiving and retention policies, auditor permissions, and other details. For more information about planning and sizing for audit and monitoring service, see the *Auditing Administrator's Guide*.

Prepare

After you have analyzed the environment, you should prepare the Active Directory organizational units and groups to use. You can then install administrative consoles and prepare initial zones.



Here are the key steps involved:

- Create organizational units or containers to define a scope of authority.

  For example, if you want to organize all of the UNIX-related information together for your organization, you can create one top-level container for the enterprise, such as Centrify UNIX. If you want to define the scope of authority at a regional or business unit level, you might have separate top-level containers for the different

regions or business units, for example, UNIX NA-SA, UNIX EMEA, UNIX PACIFIC or UNIX-Federal, UNIX-Consumer, UNIX-Industrial.

The deployment project team should consult with the Active Directory enterprise administrator to determine the appropriate top-level containers or organizational units and who should be responsible for managing and delegating administrative tasks for the objects in those top-level containers. For more information about creating organizational units or containers in Active Directory, see Designing organizational units for Centrify.

- Create the appropriate Active Directory security groups for your organization.

  Groups can simplify permission management and the separation of duties security model. For more information about using groups, see Security groups to manage Centrify information.

- Select at least one administrative Windows computer and install Centrify components Access Manager.

  This step is not strictly required if you only use existing processes or scripts to perform administrative tasks, but Centrify recommends you have at least one computer where you can use the graphical user interface to perform common tasks. If you are deploying the audit and monitoring service infrastructure, you should also install Audit Manager and Audit Analyzer. For more information about installing Centrify software on Windows, see Installing Authentication & Privilege Services.

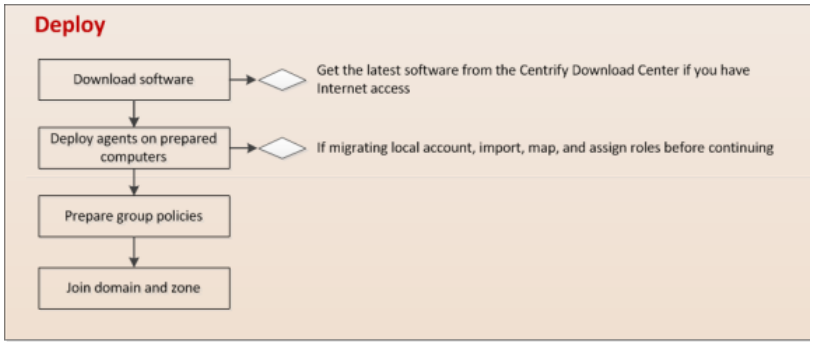
- Start the Centrify Access Manager console to run the Setup Wizard for the Active Directory domain.
- Create a parent zone and the appropriate child zones as identified in your basic zone design.

  The hierarchical zone structure you use depends primarily on how you want to use inheritance and overrides. For more information about creating parent and child zones, see Creating the first zone.

- Determine the target set of computers and make sure that they have the appropriate connectivity.

Deploy

After you have prepared Active Directory, installed administrative consoles on at least one computer, and created at least one zone, you are ready to deploy on the computers to be managed.



Here are the key steps involved:

- Download agent software from the Centrify Download Center or a network location.
- Deploy the agent software on discovered computers that are ready for installation.
- Determine whether there are any local accounts to migrate.

Right-click discovered computers, then click **Export Users and Groups** to generate a text file containing information about local accounts. Review the text file to determine whether there are any local accounts to migrate to Active Directory.

If there are local accounts that must be able to log on to the discovered computer, import the groups, then users and assign them the default UNIX Login role. For more information about migrating local accounts, see Migrating existing users to hierarchical zones.

- Join the domain using the adjoin command.
- Prepare basic group policies.

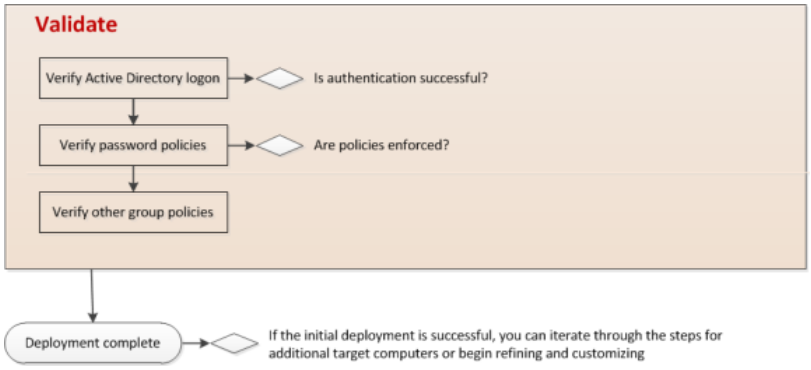
  The most common Windows computer configuration policies to deploy are:

  - Interactive Logon: Message text for users attempting to log on:—Enable and type a message that instructs the user to log on with an Active Directory user name and password.
  - Global Configuration Settings - MaxPollInterval:—Enable and set an interval if you are using Active Directory and the Centrify network time provider. Disable if you are using a native UNIX NTP daemon.
  - Enable Windows NTP Client—Enable if you are using Active Directory and the Centrify network time provider. Disable if you are using a native UNIX NTP daemon.

    The most common Centrify computer configuration policies to deploy are:
  - Set login password prompt—Enable and type a message that instructs the user to log on with an Active Directory user name and password.
  - Copy files—Enable to copy configuration files such as those required by autofs or sshd from the SYSVOL folder to managed computers.
  - Generate forwardable tickets—Disable to prevent logon tickets from being sent from one computer to another.

Validate

After you have deployed agents on target computers, you should test and verify operations before deploying on the additional computers.

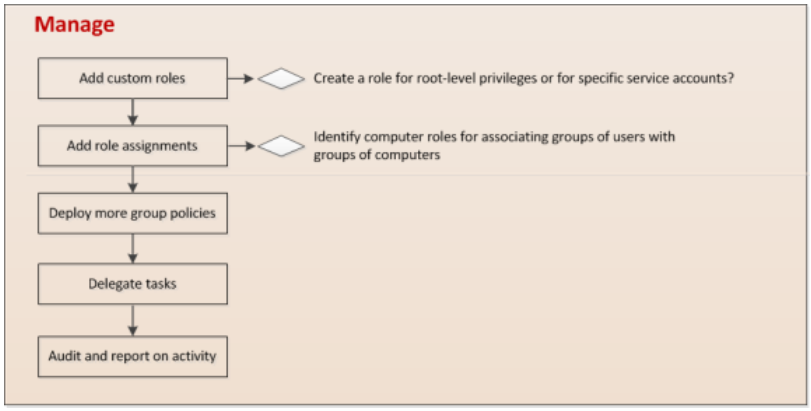


Here are the key steps involved:

- Log on to a target computer using an Active Directory user account and password to verify Active Directory authentication.
- Test password policy enforcement by attempting to change to a password that violates password complexity rules.
- Test account lockout and reset.

Manage

After you have verified the successful deployment on target computers, there are many ways you can refine, manage, and enhance on-going operations.



Here are a few of the key ways you can add value after deployment:

- Add custom roles and role assignments for users, groups, and computers.
- Import custom permissions from sudoers configuration files.
- Deploy group policies on the appropriate organizational units.
- Add the auditing infrastructure and add auditing to custom roles.
- Integrate Centrify software and Active Directory authentication and authorization services with database or web applications.

Deployment Tasks and Administrative Activity

For most deployments, there are tasks that you only perform once for an entire organization, tasks that are repeated until the deployment in complete, and tasks that are essential to deployment, but are also administrative tasks that you perform infrequently or periodically after deployment.

Steps You Only Take Once

In most organizations, you only perform the following tasks once in preparation for the deployment:

- Assemble a deployment team with Active Directory, UNIX, and other expertise.
- Provide basic training covering Centrify architecture, concepts, and terminology.
- Analyze the existing environment:

- Find a target set of computers that share a common attribute, such as the same operating system or a similar user population.
- Plan for permissions and the appropriate separation of duties for your organization.
- Review network connections, ports, firewall configuration.
- Identify computers for administration.

  Basic deployment—Access Manager

  Auditing—Audit Manager and Audit Analyzer consoles, collectors, audit databases and servers, and the installation management server

  Provisioning service—Zone Provisioning Agent and configuration tool

- Design a basic zone structure that suits your organization.
  - Single or multiple top-level parents.
  - Initial child zones, for example separate zones for Red Hat Linux and Mac OS X or different functional departments.
- Create organizational units or containers to define a scope of authority within Active Directory.
- Create Active Directory security groups for the UNIX Login role and the listed role.
- Create an Active Directory distribution group for provisioning groups and an Active Directory distribution group for provisioning users if using the provisioning service.
- Install Access Manager on at least one administrative Windows computer.
- Open Access Manager for the first time to run the Setup Wizard for the Active Directory domain.
- Create a parent zone and the appropriate child zones as identified in your basic zone design.

  Creating additional zones is an infrequent administrative task that is performed when the need arises. The basic zone design should be sufficient for the scope of your initial deployment.
- Prepare group policies to be applied.

Steps You Take More than Once During Deployment

During deployment, you perform the following tasks multiple times until you have rolled out the agent to all of the target computers that are in scope for the deployment:

- Download agent software from the Centrify Download Center or a network location.
- Deploy the agent software on computers that are ready for installation.
- If there are local accounts to migrate that must be able to log on to the discovered computer:
  - Import the groups, then users.
  - Map groups, then users to the appropriate Active Directory groups and users.
  - Assign migrated accounts the default UNIX Login role.
- Join the domain using the adjoin command.
- Verify Active Directory authentication and validate other operations.

After deployment, deploying new or updated agents is an ongoing administrative task that should be performed on a regular basis unless you have change control issues that either prevent software updates, do not allow Internet connections from the computer where Access Manager is installed, or do not want to deploy the agent on computers added to your network.

Steps You Take After Deployment to Begin Managing Zones Effectively

After you have migrated existing user populations, deployed the agent, and joined a domain, there are additional tasks you perform to complete the deployment and transition into effective zone administration.

The following tasks are optional but illustrate common administrative tasks that are often part of the deployment process to prepare for ongoing administration and improvements to operational security and efficiency:

- Create custom roles for accounts that have permission to run privileged commands.
- Create computer roles to link groups of computers with specific user role assignments.
- Map service accounts to Active Directory accounts.
- Deploy the basic set of group policies for computers and users.

What Happens After Deployment?

After deployment, ongoing management of UNIX computers, users, and groups is often handed off to Active Directory or Windows administrators or an internal service desk provisioning team to align with previously established processes and procedures for Windows servers and workstations. This is entirely a matter of organizational policy. However, in many cases UNIX administrators must continue to work with their Windows counterparts to ensure the appropriate rights and roles are assigned and the appropriate group policies are deployed.

Sample Workflow for Deployment Decisions

Centrify software solutions are extremely flexible so that they can be adapted to a wide variety of organizational requirements. All of this flexibility, however, can make deployment decisions difficult, especially in large scale or complex environments. To help you sort out the questions to ask, use the following work flow and responsibilities diagram as a guide.

This sample workflow diagram is only intended as a visual guide to the key design decisions you need to make. Many of these topics are covered in more detail in other chapters in this guide. For many organizations, however, the best guidance comes from an on-site Centrify Professional Services consultant or a Centrify partner with experience designing deployment solutions tailored to your organization’s business requirements. For customized help and advice, contact your Centrify sales representative.

Planning Organizational Units and Security Groups

One of the important steps in planning a successful deployment of Server Suite is to consider how the software fits into your Active Directory infrastructure. This section describes the issues you should consider in the planning phase that affect how Active Directory and Centrify-specific objects are organized and suggests an organizational model you can use to successfully deploy Centrify within an existing Active Directory infrastructure.

If you are planning a deployment for managing and monitoring access to Windows computers, only Licenses and Zones parent containers is applicable and you can skip the other topics in this section. If you are planning a

deployment that includes a mix of different platforms, however, you should review the recommendations for using organizational units (OUs) and groups.

If you plan to audit activity on any platform, Centrify recommends creating separate Active Directory security groups for auditors, administrators, and the computers that make up the audit and monitoring service infrastructure. Planning a deployment that includes the audit and monitoring service infrastructure requires additional resources and expertise. For more information about deploying auditing components, see the *Auditing Administrator's Guide*.

Identifying Stakeholders and Business Processes

Deploying Server Suite requires you to add objects to the Active Directory forest and, in most cases, update business processes for provisioning and removing users. It is important to identify who will be affected, which processes will be updated, how planned changes affect different parts of the business, and when you plan to deploy as early as possible in the planning stage.

It is also important to contact one or more Active Directory administrators to establish who will be creating the necessary objects in Active Directory and communicate the permissions required to create and manage those objects. If internal policies only allow Active Directory administrators to create organizational units (OUs) or security groups, you may need to negotiate when those activities take place and who will own the objects after they are created.

Identifying the appropriate people and processes early in the project helps eliminate unnecessary delays to the deployment and adoption of Centrify software. Communicating how the deployment affects the user and administrative communities helps ensure you can deploy rapidly and complete the project on-time.

The single biggest obstacle to storing UNIX data in Active Directory is overcoming internal process issues, such as change control restrictions, naming convention requirements, or proper authorization to perform administrative tasks. If you identify and resolve these challenges at the start of the project, deploying Centrify software across the enterprise becomes a fairly straightforward and painless task.

If you are a UNIX administrator, keep in mind that changes to Active Directory often require a formal change request and approval process, which can take time and delay the project. The earlier you begin planning the changes to Active Directory and the appropriate separation of duties for managing UNIX objects before, during, and after migration, the more successful the deployment will be.

Designing Organizational Units for Centrify

You can store Centrify-specific objects anywhere in the Active Directory structure if you choose. However, Centrify recommends that you create a single, high-level organizational unit (OU) specifically for Centrify objects at or near the top-level of an Active Directory forest root domain. Using one high-level organizational unit simplifies the management of Centrify containers and UNIX data.

Consolidating all UNIX data under a single organizational unit also enables you to establish an appropriate separation of duties without affecting any other previously-established OUs or permissions in Active Directory and reduces the need for additional process documentation or training. The disadvantage is that there may be strict authorization policies against setting up new organizational units.

Selecting a Location for the Top-Level OU

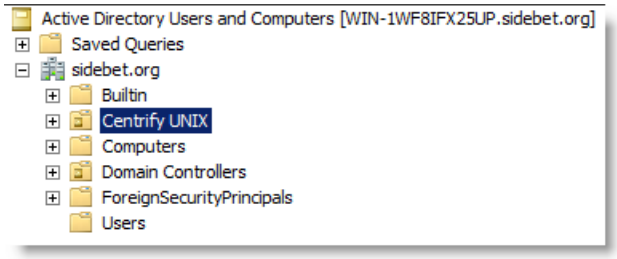
If you plan to follow the Centrify recommendation to create a top-level organizational unit for Centrify-related objects, such as Linux and UNIX computers, this high-level OU should be named so that it is easy to identify. For example, name the OU Centrify or Centrify UNIX. In deciding where this top-level OU should be placed, you should review your current Active Directory infrastructure.

There are several common scenarios:

- Single forest with a single domain
- Single forest with an empty root domain
- Single forest with account and resource domains
- Multiple forests with trust relationships
- Forests separated by a firewall (DMZ)

Single Forest with a Single Domain

If you have a single forest with a single Default-First-Site domain, you can create the top-level OU fat the same level as the default containers for Computers, Users, and other top-level objects. This is the simplest implementation of an Active Directory infrastructure. In this scenario, the top-level Centrify OU might look similar to this:



Single Forest with an Empty Root Domain

In this scenario, the forest root domain is used for the DNS namespace with one or more child domains that store information about computers, users, and groups. This is the most common Active Directory implementation. There are two important considerations if this is your Active Directory infrastructure:

- It is likely you will have a disjointed DNS namespace that you will need to resolve when you join computers to the domain.
- You might want to use multiple top-level Centrify OUs for the site. For example, assume the empty forest root domain, sidebet.org, contains three child domains:

  us.sidebet.org

  europe.sidebet.org

  asia.sidebet.org

  In this scenario, you might have a top-level Centrify OU in each child domain that includes UNIX computers to allow for more efficient data management and delegation of administrative tasks. However, if the child domains are centrally managed, you might want to create a single OU for Centrify in the forest root or one of the child

domains. In general, you should base your decision on who will be responsible for managing the Centrify objects. If there are separate administrative groups for each child domain, create a top-level Centrify OU in each child domain.

Single Forest with Account and Resource Domains

In this scenario, the forest root domain has at least two child domains. One child domain stores computer principals and related information. This domain is the **resource** domain. Another child domain stores the user and group principals. This domain is the **account** domain. This scenario requires a trust relationship that allows the computers in the resource domain to trust the users and groups in the account domain. If this is your Active Directory infrastructure, you should store the Centrify data in the resource domain. For example, if the root domain, sidebet.org, contains the child domains accounts.sidebet.org and resources.sidebet.org, you would define the top-level Centrify OU in the resources.sidebet.org (OU=Acme,DC=resources,DC=sidebet,DC=org).

Multiple forests with Trust Relationships

In this scenario, you have more than one forest needing access to Centrify data. If you have multiple forests in your organizations, you should create the top-level OU for Centrify in the forests that have UNIX computers. The forests must also be configured with either a one-way or two-way trust relationship. Cross-forest authentication requires a forest functional level of Windows Server 2003 or later. Trust relationships that involve Windows NT 4.0 domains or Kerberos V5 realms are not supported.

Cross-forest Authentication for Two-way Trust Relationships

For forests that have a two-way trust relationship, users from either forest can be authenticated to log on to the other forest. For example, if you have configured a two-way trust relationship between the forest root domain sidebet.org and youbet.org, and there are both UNIX computers and UNIX users in both forests, you would create one top-level Centrify OU in each forest and users from either forest can be authenticated to the computers in either forest.

Cross-forest Authentication for One-way Trust Relationships

To allow for cross-forest authentication with a one-way trust, Centrify authenticates users in the trusted “accounts” forest to allow those users to log on to computers in the “resources” forest. Users in the trusting “resources” forest cannot log on to computers in the “accounts” forest.

For cross-forest authentication with a one-way trust, when you add the user from the “account” forest to the Centrify zone, the user’s samAccountName attribute is stored in the zone object. Therefore, once the user is added to the zone, their samAccountName cannot change without causing authentication to fail.

Analyzing Trust Relationship to Prevent Authentication Failures

If your Active Directory environment does not permit one- or two-way trusts between forests, however, or uses a complex combination of one-way and two-way trust relationships between forests, users who attempt to log on from a remote forest may be denied access if the forest they are logging on to or the forest they are logging in from do not share a trust relationship.

As part of your deployment planning you should review your entire Active Directory infrastructure and determine whether you will be authenticating users from multiple forests and how trust relationships are defined for the forests users need access to. You may want to change the trust relationships you have defined.

For information about configuring trust relationships, see your Active Directory documentation.

Forests separated by a firewall (DMZ)

If you have a firewall between a forest outside of the firewall (the perimeter or DMZ forest) and a protected forest inside the firewall (the internal or corporate forest), the best security practice is to make the DMZ a separate forest with no trust relationship.

In this scenario, the top-level Centrify OU is created in the corporate forest protected by the firewall. This configuration ensures that the domain controllers in the perimeter forest cannot compromise the corporate domain controllers or get access to the corporate global catalog (GC), which stores information about all domains in the forest. Although defining no trust relationship between the perimeter forest and the corporate forest is considered the best practice for security reasons, this configuration prevents authentication through the firewall.

If you want to enable authentication for users in the corporate forest and allow them to access resources in the perimeter forest, Centrify recommends that you create a separate Active Directory forest for the computers to be placed in the network segment you are going to use as the demilitarized zone. You should then establish a one-way outgoing trust from the internal forest to the DMZ forest. For more information about deploying Centrify in a DMZ, see Planning to deploy in a demilitarized zone (DMZ).

Creating Recommended Organizational Units

In addition to the top-level Centrify OU, Centrify recommends you create several additional organizational units for managing UNIX groups, users, and computer accounts and rights and roles. These additional organizational units are intended to help you establish an appropriate separation of duties before, during, and after migration.

Centrify recommends the following organizational units as a starting point:

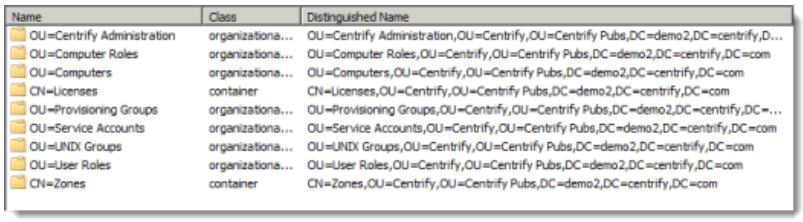
- Centrify Administration
- Computer Roles
- Computers
- Provisioning Groups
- Service Accounts
- UNIX Groups
- User Roles

Creating Organizational Units In Access Manager

If you want to create the recommended organizational unit structure for Centrify objects during your initial deployment, you can do so automatically using the Access Manager Setup Wizard. The first time you start Access Manager, it opens the Setup Wizard by default. From the Setup Wizard, you can create all of the containers for the recommended deployment structure automatically without any manual configuration. Alternatively, you can create a completely custom deployment structure by first creating a PowerShell script that creates the containers you want to use then running the custom script from within the Setup Wizard.

If you use the default script, the wizard adds the recommended organizational units and groups under the top-level **Centrify** OU and ensures all of the permissions are properly set on the objects within it. You can then select an

existing organizational unit or create a new organizational unit for the components of the deployment structure. If you use the default deployment script without modification, it creates a structure like this:



Name	Class	Distinguished Name
OU=Centrify Administration	organizationa...	OU=Centrify Administration,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,D...
OU=Computer Roles	organizationa...	OU=Computer Roles,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
OU=Computers	organizationa...	OU=Computers,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
CN=Licenses	container	CN=Licenses,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
OU=Provisioning Groups	organizationa...	OU=Provisioning Groups,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=...
OU=Service Accounts	organizationa...	OU=Service Accounts,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
OU=UNIX Groups	organizationa...	OU=UNIX Groups,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
OU=User Roles	organizationa...	OU=User Roles,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com
CN=Zones	container	CN=Zones,OU=Centrify,OU=Centrify Pubs,DC=demo2,DC=centrify,DC=com

If you want to customize the script, you can use the wizard to export it. After exporting the script, you can modify it in a text editor, then restart the wizard to use the modified script. The Setup Wizard will also create the parent container for Licenses and the parent container for Zones in the Active Directory location you select.

As you roll-out the deployment, you might find additional OUs are useful. For example, you might create additional OUs because specific permissions must be granted to create, modify, delete, and manage the objects within them. By creating this organizational structure, you can control who has permission to manage the Centrify objects contained in each OU.

By using the recommended deployment structure and associated permissions, you will have a solid foundation for deploying Centrify software across the enterprise without affecting any of your existing Active Directory structure. The recommended deployment structure will also enable you to easily apply group policies and manage user, group, and computer accounts.

For more information about the purpose of the additional organizational units, see the appropriate section.

Centrify Administration Organizational Unit

The Centrify Administration OU is intended to store Active Directory security groups that ensure the separation of duties and the segregation of Centrify-related administrative operations in Active Directory.

In most cases, you will want to allow your existing Active Directory account fulfillment or provisioning team to edit UNIX groups and, potentially, the user role groups that allow for elevated permissions in UNIX. If users you have identified as Centrify Administrators are stored in the same organizational unit as the rest of the UNIX groups, then members of the fulfillment or provisioning team could grant themselves permissions to create, modify, and delete zones. With these permissions, a disgruntled provisioning staff member could delete one or more zones and prevent access to production computers. To prevent this security risk, Centrify recommends you create a separate Centrify Administration organizational unit and protect access to it.

Computer Roles Organizational Unit

The Computer Roles OU is intended to store the computer group accounts that are associated with a specific computer role. For example, if you plan to have a computer role for computers that host Oracle databases and the set of users assigned the database administrator role, you might create an Active Directory security group called Oracle_Production_Computers in this OU for the computers that host Oracle databases. If you were to add a new Oracle database instance, you would add the computer account for that database server to the Oracle_Production_Computers in this OU.

In most cases, the computer groups in this organizational unit are associated with the user role groups you add to the User Roles OU. For example, if you have a computer role for computers that host Oracle databases, you might have user role groups for database administrators and another user role group for database users. If you were to

change who should be allowed to use the database or perform database administration activities, you would modify the membership of these two user role groups.

Computers Organizational Unit

The Servers OU is intended to store new computer principals in Active Directory. This organizational unit enables you to efficiently deliver computer-based group policies from Active Directory. For example, you can use a group policy to turn off SNTP updates from Active Directory for Centrify-managed computers to prevent those computers from having two registered time sources. Having a separate OU also enables you control who has the permissions and authority to create, update, and delete computer objects in the domain.

Provisioning Groups Organizational Unit

The Provisioning Groups OU is intended to store Active Directory distribution groups that are used by the Zone Provisioning Agent. The Zone Provisioning Agent is a Windows service that processes the business rules for creating or deleting UNIX profiles in zones. For example, if a new Active Directory user principal is in one of these group principals, and the group is associated with a zone, the user is automatically provisioned with a UID and a GID in that zone.

> The profile does not allow the user to log on to computers in the zone. Identity management is separate from access management. The user's role assignments control access.

During the migration process, users you have identified as Centrify Administrators should have the appropriate permissions and authority to create, delete, and manage the membership of these Active Directory distribution groups. After migration, the team that owns the process for the provisioning UNIX accounts will need the same permissions and authority. For details about the permissions required to perform these tasks, see Setting permissions for zone groups.

Service Accounts Organizational Unit

The Service Accounts OU is intended to store the Windows service account for the Zone Provisioning Agent and any UNIX-specific service accounts that do not correlate to existing Active Directory user accounts. For example, you can use this OU for application service accounts, such as Oracle or MySQL, to enable centralized password management and auditing. By migrating service accounts from UNIX, you can centrally manage the account information, passwords, and privileges for those service accounts and their associated UNIX groups.

Unix Groups Organizational Unit

In this deployment model, the UNIX Groups OU is intended to store Active Directory security groups that are migrated from /etc/group files or NIS group maps that you want to preserve.

As part of the initial migration, you should identify one or more users as **Centrify Administrators** who should be granted the appropriate permissions and the authority to create new Active Directory group principals, and to add Active Directory user principals to the new groups.

After the migration is complete, another team might be responsible for managing the UNIX groups migrated into Active Directory. The team that owns the process for adding UNIX users to a UNIX group, removing UNIX users from a UNIX group, or creating new UNIX groups will need the permissions and the authority in Active Directory to create and delete group principals and manage the membership of those group principals in this OU (for example,

ou=unix groups,ou=Centrify). For details about the permissions required to perform these tasks, see Setting permissions for zone groups.

User Roles Organizational Unit

The User Roles OU is intended to store Active Directory security groups that are associated with user role definitions that grant privileges or restrict access. For example, a user role definition might grant permission to execute commands as root or using a service account such as oracle. By associating an Active Directory group with a role definition, you can grant or deny privileges by managing Active Directory group membership.

During the migration process, users you have identified as Centrify Administrators should have the permissions and the authority to add users to the appropriate user role groups and to create new Active Directory group objects in the OU (for example, ou=user roles,ou=Centrify). After migration, your organization should decide who should be responsible for creating new user role groups and associating them with zones and who should be able to add and remove users from the User Roles organizational unit.

Licenses And Zones Parent Containers

Regardless of whether you choose to create the organizational units for the recommended deployment structure or a custom deployment structure, Centrify requires the following parent containers:

- Licenses parent container object for license keys. You must have at least one parent container for license keys in the forest. You can create more than one of these container objects to give you more granular control over who has access to which licenses.
- Zones parent container object for individual zone (ZoneName) objects. You must have at least one parent container for zones. You can create more than one parent Zones container to give you more granularity for delegating administrative tasks.

You can select the parent containers for Licenses and Zones when you run the Setup Wizard, when creating a new zone, or when managing licenses in Access Manager.

Some organizations prefer to create and manage Active Directory objects manually to ensure tight control over the objects and their attributes. For example, you might want to manually create separate parent containers for different business departments or locations if you want to manually set permissions and refine who has access to them. However, managing permissions manually can be complex and error-prone. In most cases, Centrify recommends that you establish appropriate permissions on the deployment structure and use the Zone Delegation Wizard to manage administrative permissions on individual zones and the objects contained in zones.

Security Groups To Manage Centrify Information

If you use the default recommended deployment script, the script automatically creates the following Active Directory security groups for managing Centrify-related objects:

- CentrifyAdministrators
- AuthorizationManagers
- ComputerManagers
- UnixDataManagers

Each security groups is granted the appropriate permissions to perform specific administrative tasks. For example, users who are members of the Centrify Administrators group should be able to create, modify, and delete zones.

If you are not using the recommended deployment script, you should create similar security groups for managing Centrify-related objects.

Delegating Control For Centrify Administrators

The CentrifyAdministrators security group is intended for members of the administrative or security team who are responsible for managing all Centrify-related information stored in Active Directory. You should add members to the group to grant specific users the rights required to manage Centrify licenses, zones, user roles, computer roles, provisioning groups, and the user, group, and computer profiles in each zone.

Members of the Centrify Administrators security group are responsible for identifying an organization's zone requirements and creating the zone hierarchy. Centrify Administrators also decide when to create new zones, delete obsolete zones, or consolidate existing zones. In most cases, Centrify Administrators define the basic access rights for zones and delegate administrative tasks to other users and groups on a zone-by-zone basis.

Permissions for the Centrify Administrators group are applied at the top-level of the deployment structure—for example, ou=Centrify—and grant privileges on the organizational units, containers, and object within the deployment structure. If you don't create this group or a similar security group, only members of the Domain Admins group can create new zones.

If you are managing security and individual permissions manually for Active Directory objects, see Permissions required for administrative tasks for information about the permissions required for individual tasks.

Delegating Control For Authorization Managers

The AuthorizationManagers security group is intended for members of the security team who are responsible for managing role-based access rights. You should add members to the group to grant specific users the rights required to manage user roles, computer roles, access privileges, and role assignments.

You can delegate tasks to the AuthorizationManagers group on the User Roles and Computer Roles organizational units using Active Directory Users and Computers. You can delegate zone administration tasks to the group in Access Manager.

Delegating Tasks For User Role Groups

In Active Directory Users and Computers, select the User Roles organizational unit, right-click, then select Delegate Control to start the Delegation of Control Wizard. Select the security group you are using for authorization managers and delegate the following tasks:

- Create, delete and manage groups
- Modify the membership of a group

Delegating Tasks For Computer Role Groups

In Active Directory Users and Computers, select the User Roles organizational unit, right-click, then select Delegate Control to start the Delegation of Control Wizard. Select the security group you are using for authorization managers and delegate the following tasks:

- Create, delete and manage groups
- Modify the membership of a group

Delegating Zone-specific Tasks

As a member of the CentrifyAdministrators security group, you can grant zone-specific permissions to the members of the AuthorizationManagers group. After you have created the appropriate zones, you can delegate the following zone administration tasks to authorization managers:

- Manage roles and rights
- Manage role assignments
- Modify computer roles
- Add computer roles

Delegating Control For Computer Managers

The ComputerManagers security group is intended for members of the UNIX administration team who are responsible for managing computer accounts. You should add members to this security group to grant specific users the rights required to manage computer objects in the Servers organizational unit in Active Directory.

As a member of the CentrifyAdministrators security group, you can also grant zone-specific permissions to the members of the ComputerManagers group. After you have created the appropriate zones, you can delegate the following zone administration tasks to computer managers:

- Join computers to the zone
- Remove computers from the zone

If you are managing security and individual permissions manually for Active Directory objects, see Permissions required for administrative tasks for information about the permissions required for individual tasks.

Delegating Control For Unix Data Managers

The UnixDataManagers security group is intended for members of the UNIX administration team who are responsible for managing computer accounts. You should add members to this security group to grant specific users the rights required to manage UNIX users and groups objects in the UNIX groups and Service Accounts organizational units in Active Directory.

You can delegate tasks to the UnixDataManagers group on the UNIX Groups and Service Accounts organizational units using Active Directory Users and Computers. You can delegate zone administration tasks to the group in Access Manager.

Delegating Tasks For Unix Groups

In Active Directory Users and Computers, select the UNIX Groups organizational unit, right-click, then select Delegate Control to start the Delegation of Control Wizard. Select the security group you are using for UNIX data managers and delegate the following tasks:

- Create, delete, and manage groups
- Modify the membership of a group

Delegating Tasks For Service Accounts

In Active Directory Users and Computers, select the Service Accounts organizational unit, right-click, then select Delegate Control to start the Delegation of Control Wizard. Select the security group you are using for UNIX data managers and delegate the following tasks:

- Create, delete, and manage user accounts
- Reset user passwords and force password change at next logon

Delegating Zone-Specific Tasks

As a member of the CentrifyAdministrators security group, you can also grant zone-specific permissions to the members of the UnixDataManagers group. After you have created the appropriate zones, you can delegate the following zone administration tasks to UNIX data managers:

- Add users
- Add groups
- Remove users
- Remove groups
- Modify user profiles
- Modify group profiles

Planning for Data Storage in Active Directory

Centrify stores all of the UNIX attributes required for users, groups, and computers in Active Directory, so that this information can be centrally managed. It stores these attributes without requiring you to make any modifications to the Active Directory schema you choose to use.

Changing The Zone Type

If you create a new zone using the default zone options, the new zone is created as a hierarchical zone that uses the Active Directory RFC2307-compatible schema attributes for user and group profiles. If you deselect the Use the default zone type option, you can choose to create either a hierarchical zone or a classic zone and how you want zone information stored in the Active Directory schema.

If you are not using the default zone type and storage model, you have the following options:

- A **Standard zone** stores user and group attributes in the keywords attribute of the serviceConnectionPoint object for the user or group rather than in the user or group object.
- An **RFC2307-compatible zone** stores user and group attributes in the attributes that are defined in the RFC2307-compatible schema for user and group objects.
- An **SFU zone** stores user and group attributes in the Services for UNIX (SFU) schema attributes for the user or group object.

It is worth noting that in the default zone storage model—which uses the default Active Directory RFC2307-compatible schema—some schema attributes are not indexed. For example, in the default Active Directory

RFC2307-compatible schema, the uid attribute is not an indexed attribute. Because of this limitation, queries that use this attribute might take longer than expected.

Modifying Indexed Attributes For Zones

Depending on the requirements of your organization, you might see improved performance either by creating standard Centrify zones rather than RFC2307-compatible zones or by indexing the uid attribute in default zones. Before selecting a strategy for all or selected zones, however, you should consider that indexing the uid attribute requires you to modify the Active Directory schema.

Modifying the Active Directory schema is an advanced operation that should only be performed by experienced administrators. In addition, you must be a member of the Domain Admins group or the Enterprise Admins group in Active Directory or been delegated similar authority to perform this operation. If you choose to modify the schema to improve performance, you can use “Run as ...” to select an account with appropriate permissions before performing the following steps.

To index an attribute:

1. Open a Command Prompt window, then type the following command to register the schema management assembly on your computer:

   regsvr32 schmmgmt.dll
2. Click Start, click Run, type mmc /a, then click **OK**.
3. In the console root, select the File menu, then click **Add/Remove Snap-in**.
4. Under Available Standalone Snap-ins, double-click Active Directory Schema, click **Add**, then click **OK**.
5. On the File menu, click **Save**, navigate to the %systemroot%/Sytem32 directory, type a file name for the console, then click **Save**.
6. Click **Start > All Programs** to select the Administrative Tools folder, right-click, then select **Open**.
   - If necessary, select **Organize > Layout > Menu bar** to display menus.
   - On the File menu, select **New**, then click **Shortcut**.
   - In the Create Shortcut Wizard, click in **Type the location of the item**, type the name you used for the file in Step 5, then click **Next**.
   - Select the file name in Type a name for this shortcut field, type **Active Directory Schema**, then click **Finish**.
7. Click **Start > All Programs > Administrative Tools** to select Active Directory Schema, right-click, then select **Open**.
8. Expand Attributes to select a specific attribute, such as uid, right-click, then select **Properties**.
9. Select **Index this attribute**, then click **OK**.

Viewing and Manipulating Data in Active Directory

You can view, access, and manage any information stored in Active Directory—including Centrify profiles, rights, roles, and role assignments—using ADSI Edit or using any tools that can perform standard LDAP operations such as ldifde and OpenLDAP commands such ldapsearch, ldapadd, ldapdelete and ldapmodify. For example, depending on the type of operating system and tools you prefer to use, you might view and manage Centrify profiles and zones using any combination of the following tools:

- Access Manager
- Access Module for Windows PowerShell
- Audit Module for Windows PowerShell
- Active Directory Users and Computers
- The Server Suite Windows API
- The ADEdit Tcl application and procedure library
- Centrify command-line programs

By using these tools, you can manipulate Centrify information manually or create scripts to automate key tasks such as the provisioning of new accounts. For example, you can write scripts that access the Centrify Windows API or ADEdit procedures to automatically create computer, user, or group accounts, create new zones, or assign users to roles. As part of your planning process, you should determine whether there are tasks you want to automate through the use of scripts, so that members of the development team can create or modify the appropriate tools and test them thoroughly before deploying across the organization.

Installing Authentication & Privilege Services

This section provides instructions for installing all identity and privilege management components on Windows computers in your network. There are several Windows-based components that enable you to manage the deployment and ongoing operations of Server Suite software. You should install all of the identity and privilege management components on at least one Windows computer. Depending on the division of responsibilities in your organization, you may want to install different components on more than one Windows computer.

When you install identity and privilege management components, the following features are installed:

- The Privileged Access Service, which enables MFA login, MDM, and other platform services.
- The Privilege Elevation Service and Authentication Service, which together enable computers where Server Suite software is installed to use the Active Directory infrastructure located on the domain controller, and enable users and zone-joined computers to have elevated privileges. The services include ADUC extensions, GPOE extensions, PowerShell extensions, Server Suite utilities, and Access Manager.

Access Manager is the administrative console that enables you to create zones and configure rights and roles for Active Directory users running applications on Windows computers.

You should always install the Windows components first before you install the Server Suite Agent on the non-Windows computers you intend to manage.

Preparing for Installation on Windows

Before installing Server Suite management components on Windows, you should verify that the computers where you are planning to install meet all of the system requirements and prerequisites and that you have all of the information you need to install and configure the software packages.

At a minimum, you should install the following Server Suite components on one or more Windows computers during the first stage of deployment:

- Access Manager console
- Zone Provisioning Agent

You can install these components together or independently using the setup program. Alternatively, you can install these components independently without running the setup program by using individual setup programs for each component.

Installing Server Suite

Access Manager, which is installed when you install Server Suite, is the primary management console for performing access control and privilege management operations. You typically install Access Manager directly on the computers used by one or more administrators. Alternatively, you can install it on a physical or virtual server accessed remotely by one or more administrators. The most important requirement is that the computer where you install Access Manager must be able to connect to the Active Directory domain and forest.

The Access Manager console can be installed from the setup program or from a standalone executable separate from the setup program. Before you install, you should verify your environment meets the system requirements to ensure a successful deployment.

Preparing Active Directory and DNS

All of the Server Suite software components rely on critical pieces of Active Directory infrastructure. Before you install:

- Verify Active Directory is installed and you have access to at least one Windows computer acting as a domain controller for the Active Directory forest to which you want to add UNIX computers.
- Check the configuration of DNS and whether you are using a Windows computer as the primary DNS server.
- Verify the DNS server allows secure dynamic updates and your domain controllers are configured to publish updated service locator (SRV) records.
- Verify DNS resolution and network communication between the UNIX computers and the Active Directory domain controller. You can use the ping command to test communication between the domain controller and the UNIX computer.

Identifying the Windows Computer and Log On Credentials

Depending on how you plan to manage Server Suite properties, you should identify an appropriate Windows computer and the user account credentials you should use. For example:

- Check whether the Windows computer has Active Directory Users and Computers installed.

  If you want to manage Server Suite properties using Active Directory Users and Computers, the Active Directory Users and Computers MMC snap-in must be available on the local computer.
- Check whether the Windows computer is a domain computer, such as a Windows XP workstation, or a domain controller.

  If you install on a domain controller, you must use your own logon credential to connect to Active Directory. In most cases, you can install on any computer that has access to a domain controller.
- Verify that the Windows computer can connect to Active Directory.

- Verify that you have a Windows user account and password with sufficient rights to install software on the local computer and permission to update the Active Directory forest.

  After installation, you must be able to create new container objects in the Active Directory forest. Alternatively, an Active Directory administrator can manually configure the environment or temporarily modify your account permissions to enable you to perform setup tasks. For information about the specific rights required to perform tasks in the Setup Wizard, see Permissions required to use the Setup Wizard.

Checking Operating System and Software Requirements

Before installing on Windows, check that you have a supported version of one of the Windows operating system product families. For example, you can use Windows 7 for any console components. Alternatively, you can install components on computers in the Windows Server product family–such as Windows Server 2008 or Windows Server 2012–so that your administrative computer can be configured with additional server roles.

For more detailed information about supported platforms for specific components, see the release notes.

You should also verify that you have the .NET Framework, version 4.5 or later, installed. If the .NET Framework is not installed, the setup program can install it for you. Alternatively, you can download the .NET Framework from the Microsoft Download Center, if needed.

Checking Disk and Memory Requirements

You should also check that the computer where you are installing the Access Manager console meets the following requirements:

For this	You need this
CPU speed	Minimum 550 MHZ
RAM	25MB
Disk space	100MB

Running the Setup Program on a Windows Computer

You can install Server Suite software using the setup program on the CD or included in the download package. The setup program copies the necessary files to the local Windows computer. There are no special permissions required to run the setup program other than permission to install files on the local computer. From the setup program, you can choose which components of you want to install.

> If you intend to install the Zone Provisioning Agent using the setup program, you should review the requirements and other information in Installing Zone Provisioning Agent before you proceed, but you can skip the standalone installation instructions in those sections. Use the individual setup programs for components if you want to install a specific component on a specific computer. For example, use the Centrify_Zpa*version*win64.exe program to selectively install Zone Provisioning Agent components on a computer where Access Manager is not installed.

To install Authentication & Privilege on Windows:

1. Log on to the Windows computer and insert the CD or navigate to the directory where you downloaded Server Suite files.

   If the Getting Started page is not automatically displayed, double-click the autorun.exe program to start the installation of the Server Suite software.

2. On the Getting Started page, click **Authentication & Privilege** to start the setup program for identity and privilege management components.

   If any programs must be updated before installing, the setup program displays the updates required and allows you to install them. For example, you might be prompted to install or update the Microsoft .NET Framework or Microsoft SQL Server Compact edition.

3. At the Welcome page, click **Next**.

4. Review the terms of the license agreement, click **I agree to these terms**, then click **Next**.

5. Type your name and organization, then click **Next**.

6. Expand and select the Delinea Administration and Delinea Utilities components you want to install, then click **Next**.

   If you are managing access to Linux, UNIX, and Mac OS X computers, you should select the following Server Suite Administration components for deployment:

   - **ADUC property page extensions** if you want to include Server Suite profiles when displaying properties in Active Directory Users and Computers.
   - **Access Manager** if you want to use an administrative console to manage Server Suite zones and roles.
   - **Group Policy Management Editor extension** if you want to deploy Server Suite group policies.

     You should also select the following Server Suite Utilities components for deployment:

   - **Zone Provisioning Agent** if you want to automatically provision user and group profiles into zones.

     If you want to skip the installation of any component on the local computer, click to deselect the item that you want to skip, then click **Next**. For example, if you want to skip installation of the Server Suite Reporting Service and its Microsoft SQL Server database, deselect the Server Suite Reporting Service option, then click **Next**.

7. Accept the default location for installing components, or click **Browse** to select a different location, then click **Next**.

8. Review the components you have selected, then click **Next**.

   The setup program begins installing the selected components.

9. When setup is complete for the selected packages, click **Finish** to close the setup program.

   Depending on the components you selected, you might see options to configure reporting service, the Zone Provisioning Agent, or both. You can deselect these options if you want to skip configuration or plan to install the components in a different computer. For details about configuring the Server Suite reporting service, see the *Report Administrator's Guide*. For details about configuring the Zone Provisioning Agent after installing it with the Server Suite setup program, see Configuring the Zone Provisioning Agent.

Installing Zone Provisioning Agent

The Zone Provisioning Agent enables automated provisioning of user and group accounts into Server Suite zones. You configure the Zone Provisioning Agent to monitor specific Active Directory groups that are linked to a zone. When you add or remove users or groups from the monitored groups, the Zone Provisioning Agent adds or removes corresponding users or groups in the zone.

You can install the Zone Provisioning Agent with the Server Suite setup program or as a standalone service separate from the installation of other Server Suite components. In most cases, it is installed on its own apart from the installation of other Server Suite components. After the Zone Provisioning Agent is installed, you can configure the business rules for adding and removing groups and how the attributes associated with user or group profiles are automatically generated.

About Zone Provisioning Agent and its Requirements

The Zone Provisioning Agent is intended to run on an ongoing basis on a computer that is always available. It requires a Windows user account with the right to Log on as a service. If you have a single forest, you can install the Zone Provisioning Agent on one or two computers. If you install the Zone Provisioning Agent on two computers, you should only run one instance at a time. The Zone Provisioning Agent on the second computer is intended for standby operation. You should only start the Zone Provisioning Agent on the second computer if the first instance fails.

> The business rules that control provisioning are stored in Active Directory. If only one computer has the Zone Provisioning Agent and that computer stops running, the automated provisioning of UNIX users and group is interrupted until the computer and the Zone Provisioning Agent are restarted. Users with existing access to UNIX computers are not affected.

The Zone Provisioning Agent has the following components:

- **Zone Property Page Extension** must be installed on the same computer as the Access Manager console. This extension adds a tab to the Zone Properties for configuring provisioning rules.
- **Provisioning Agent** can be installed separately from the property page as a standalone service or on the same computer as Access Manager. The computer where you install the service should be available at all times. In most cases, this Windows service is not installed on the same computer as Access Manager.
- **Command Line Utility** can be installed separately or on the same computer as Access Manager. The command line utility allows you to write scripts for provisioning tasks or update zones on demand.

If you have more than one forest, you should install a Zone Provisioning Agent in each forest. If you have geographical domains within a single forest, you may want to install a Zone Provisioning Agent in each geographical domain. If you install a second instance of the Provisioning Agent for failover, be sure that only one instance of the Provisioning Agent runs in each forest.

Zone Provisioning Agent account permissions

Account name (suggested)	Type of account	Required permissions	Notes

Cfy_SVC_ ZPA	Active Directory account	**Log on as a service**	The Zone Provisioning Agent requires permission to create UNIX profiles-- that is, the service connection points in each zone where it needs to perform provisioning operations. The service account that runs the Zone Provisioning Agent requires the **Log on as a service** right set as a local computer security policy, or in the default domain policy.

Create a service account for the Zone Provisioning Agent

The Zone Provisioning Agent must run using a valid Windows user account with the right to Log on as a service. In most cases, you should create a dedicated user account, called the **service account**, for the service to run as rather than use an existing user account.

To create a new service account for the Zone Provisioning Agent:

1. Open Active Directory Users and Computers.
2. Select the **UNIX Service Account** organizational unit.
3. Right-click, then select **New > User**.
4. Type a display name and logon name for the service account, then click **Next**.
5. Type and retype a password for the service account and modify the account options as follows, then click **Next**:
   - Uncheck **User must change password at next logon**
   - Check **User cannot change password**
   - Check **Password never expires**
6. Click **Finish** to add the service account.

Configure the local or domain group policy to allow the account to log on as a service

After you have created the service account, you must edit either a local security policy or the default domain group policy to grant the service account the **Log on as a service** right.

If you edit the default domain policy, the Zone Provisioning Agent can run on any Windows computer. If you need to move the service from one computer to another, no additional configuration is required.

Alternatively, you can edit the local security policy specifically on the computers that run the Zone Provisioning Agent. If you use the local policy, however, you may need to investigate whether other group policies are applied to the computer running the Zone Provisioning Agent to see if inheritance disables your local policy setting.

To edit the default domain group policy:

1. Open the Group Policy Object Editor and navigate to **Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment > Log on as a service**.
2. Right-click **Log on as a service**, then select **Properties**.
3. Select **Define these policy settings**, then click **Add User or Group**.
4. Click **Browse** to search for the service account you created.
5. Select the service account, then click **OK** to add the account and **OK** again to apply the policy.

Installing the Zone Provisioning Agent on the Access Manager computer

You can install both the Zone Provisioning Agent service and the Zone Property Page Extension on the computer where Access Manager is installed. At a minimum, you should install the Zone Property Page Extension on the same computer as the Access Manager console. The Zone Property Page Extension enables you to configure the Active Directory groups to monitor and the business rules for how to derive each user and group attribute.

> If you select the Zone Provisioning Agent when you install components using the Server Suite setup program, all Zone Provisioning Agent components are installed. If you want to selectively install Zone Provisioning Agent components on a computer, you can install by running the Centrify_Zpaversionwin64.exe program.

To install the Zone Provisioning Agent on the Access Manager computer:

1. Log on to the Windows computer where Access Manager is installed.
2. Double-click the Centrify_Zpaversionwin64.exe file to start the Zone Provisioning Agent setup program.
3. If a User Account Control message is displayed, click **Yes**.

   If necessary, the setup program prompts you to install the Delinea Common Components.
4. On the **Welcome** screen, click **Next**.
5. Accept the licensing agreement, then click **Next**.
6. Select the features to install, then click **Next**.

   The Zone Property Page Extension is only applicable on the computer where the Access Manager console is installed. Selecting this option adds the Provisioning tab to the Zone Properties for individual zones. You can install or uncheck the other features on the computer where the Access Manager console is installed.
7. Click **Next** to accept the default location for the Zone Provisioning Agent files, or click **Browse** to select a different location, then click **Next**.
8. Click **Install** to begin installation.
9. Click **Finish** to complete the installation.

Installing the Zone Provisioning Agent on its own

You can install the Provisioning Agent as a standalone service on a computer with a relatively light load. The computer where you install the Provisioning Agent should be one that is online at all times. If the computer is shut down or suspended, the Provisioning Agent service will be suspended and no provisioning can occur. You can install a second instance of the Zone Provisioning Agent on another computer, and use your existing method of determining if a service has failed to monitor the availability of the first instance. For example, configure monitoring of the Windows Event log to notify you if the Zone Provisioning Agent service stops.

If you install the Provisioning Agent service as a standalone service, you should also install the Command Line Utility on the same computer.

To install the Zone Provisioning Agent as a standalone service:

1. Log on to the Windows computer that has a light load and is rarely shut down or offline.
2. Double-click the Centrify_Zpaversionwin64.exe file to start the Zone Provisioning Agent setup program.
3. If a User Account Control message is displayed, click **Yes**.

   If necessary, the setup program prompts you to install the Delinea Common Components.
4. On the **Welcome** screen, click **Next**.
5. Accept the licensing agreement, then click **Next**.
6. Select the Provisioning Agent and Command Line Utility features, then click **Next**.
7. Click **Next** to accept the default location for the Zone Provisioning Agent files, or click **Browse** to select a different location, then click **Next**.
8. Click **Install** to begin installation.
9. (Optional) Uncheck the **Configure and start Zone Provisioning Agent** option, then click **Finish**.

   If you leave **Configure and start Zone Provisioning Agent** selected, you are prompted to provide the service account name and password, then click **Start** to start the agent service. It is recommended that you configure the monitored containers, polling interval, and logging options, in addition to the service account name and password before starting the service. Therefore, you should open Access Manager to set up the Server Suite organization structure in Active Directory. For more information about the initial configuration, see Running Access Manager for the first time.

Configuring the Zone Provisioning Agent

By default, the Zone Provisioning Agent monitors all domains in the entire forest. If you use the recommended Server Suite organizational structure described in Creating recommended organizational units, it is recommended setting the Zone Provisioning Agent to only monitor the top-level Server Suite organizational unit or the Zones container. These objects are created in the Setup Wizard the first time you open Access Manager. After the initial configuration, you can perform the steps in this section to configure the Zone Provisioning Agent. For more information about the initial configuration, see Running Access Manager for the first time.

The most common reason for monitoring more than one organizational unit is if you have a regional or team-based OU structure in Active Directory, where each region or team is responsible for managing its own UNIX data. In this scenario, a provisioning staff member in Sydney, Australia, wouldn't be responsible for account fulfillment of a UNIX user in Chicago. To ensure the appropriate separation of duties between the different regions or teams, you would have more than one Server Suite organizational unit, and you would configure the Zone Provisioning Agent to search each of the regional organizational units.

To configure the Zone Provisioning Agent

1. Open the Zone Provisioning Agent Configuration Panel by clicking **Start > All Programs > Server Suite 2021.1 > Zone Provisioning Agent Configuration Panel**.
2. In the Monitored containers section, click **Add**.
3. Navigate to select the Server Suite organizational unit or the Zones container, then click **OK**.
4. Select **Entire Forest***forest_name* from the list of Monitored containers, then click **Remove**.
5. Set the provisioning polling interval in minutes.

The polling interval controls how often the Zone Provisioning Agent checks monitored containers for changes and processes the business rules for provisioning users and groups into zones. The appropriate interval often depends on the expectations of the user population or on service level agreements that define the provisioning team's commitments. In general, you should avoid polling more frequently than necessary to reduce the affect the Zone Provisioning Agent has on the performance of your domain controllers.

6. If desired, you can specify which domain controller that the Zone Provisioning Agent uses.
   a. To specify the domain controllers, click **Advanced**.

      The Advanced Domain Controller Settings dialog box displays.

   b. Click **Add** to open a separate dialog box in which you can add a domain and pick from a list of domain controllers. Click **OK** to save your chances.

   c. Click **Change** if you want to change the specified domain controller, or click **Remove** if you need to remove the specified domain controller.
7. Type the service account name or click **Browse** to locate the service account name, then type the password for the account.
8. Click **Apply**.
9. Click **Start** to start the Zone Provisioning Agent.

Whitelisting Domains for the Zone Provisioning Agent

You can configure the Zone Provisioning Agent so that it can connect to trusted domains (whitelisting) by setting the following registry key with a list of trusted domains and/or forests:

HKLM\SOFTWARE\Centrify ZPA\AllowedDomains

Configuring a list of domains this way can be particularly useful and faster when you have a large amount of domains.

For example, to specify a single domain:

HKLM\SOFTWARE\Centrify ZPA\AllowedDomains: "acme.com"

For example, to specify multiple domains:

HKLM\SOFTWARE\Centrify ZPA\AllowedDomains: "acme.com", "foo.com"

Running Access Manager for the First Time

The first time you start the Access Manager console, a Setup Wizard guides you through the initial configuration of the Active Directory forest. This initial setup creates the recommended or a custom deployment structure including the parent containers for Licenses and Zones and sets the permissions for modifying the objects within the containers. These steps are only performed once and can be done manually, if you choose.

Because the Setup Wizard creates container objects, you might need to use a domain administrator account. This requirement depends on the specific permissions your organization has configured for different classes of users. For example, if your organization only permits Domain Admins to create parent and child objects in Active Directory, you need to use an account with those permissions to run the Setup Wizard. For more information about the permissions required to perform specific configuration steps, see Permissions required to use the Setup Wizard.

Access Manager Account Permissions

Account name (suggested)	Type of account	Required permissions	Notes
n/a	Domain administrator (when running Access Manager for the first time)	domain admin (in most cases)	Because the Setup Wizard creates container objects, you might need to use a domain administrator account. This requirement depends on the specific permissions your organization has configured for different classes of users. For example, if your organization only permits Domain Admins to create parent and child objects in Active Directory, you need to use an account with those permissions to run the Setup Wizard.

To start the Setup Wizard and update the Active Directory forest

1. Open Access Manager from the desktop shortcut or Start menu.
2. Verify the name of the domain controller displayed is a member of the Active Directory forest you want to update or type the name of a different domain controller if you want to connect to a different forest, then click **OK**.
   - If you want to connect to a different forest, type the name of a domain controller in that forest.
   - If you want to connect to the forest with different credentials, select **Connect as another user**, then type a user name and password to connect as.
3. At the Welcome page, click **Next**.
4. Select **Use currently connected user credentials** to use your current log on account or select **Specify alternate user credentials** and type a user name and password, then click **Next**.
5. Select **Generate the recommended deployment structure** if you want to create all of the containers for the recommended deployment structure automatically.

   If you select this option, select whether you want to generate the default deployment structure or generate a custom structure, then click **Next**.
   - If you are generating the default structure, clicking Next enables you to select or create the location for the deployment structure in Active Directory. For example, if you want to create the top of the default deployment structure at the domain level, click **Next**, then click **Browse** to select the domain name. After you have selected a location, click **OK**. then click **Next** to create the deployment structure.
   - If you are generating a custom structure, clicking Next enables you to export the script that creates the default structure or run a script you have previously written.

     If you are generating a default or custom deployment structure, verify the successful execution of the script that creates the structure, then click **Next** to continue.
6. Verify the parent container for licenses is in the top-level Server Suite container if you are using the default deployment structure or the container of your choice, then click **Next**.

   You can add other Licenses containers in other locations later using the Manage Licenses dialog box.

If you are not using the recommended deployment structure, the default container for license keys is domain_name/Program Data/Centrify/Licenses. To create the parent container in a different location, you can click **Browse**.

7. Review the permission requirements for the container, then click **Yes** to continue.

   If you don't want to allow the permissions for the selected container, click **No** and select a different container to continue.

8. Type or copy and paste the license key you received, then click **Add**.

   If you received multiple license keys, add each key to the list of installed licenses, then click **Next**. If you received license keys in a text file, click **Import** to import the keys directly from the file instead of adding the keys individually, then click **Next**.

   You can also add and remove license containers and keys after the initial configuration.

   For details about licensing, including how to request new license keys after deployment, check license usage and compliance, and how license counts are determined, see the *License Management Administrator's Guide*.

9. Verify that the **Create default zone container** option is selected and the parent container for zones is in the top-level Server Suite container or the container of your choice, then click **Next**.

   If you are not using the recommended deployment structure, the default container for zones is domain_name/Program Data/Centrify/Zones. To create the parent container in a different location, you can click **Browse**.

   You can skip creating the parent container in the forest or have more than one Zones parent container. For example, if you have a regional OU structure in Active Directory—where each region is responsible for its own set of zones—each region should have its own top-level organizational unit. For example, if you have separate OU structures for Tucson, AZ, and Newark, NJ, you would have separate deployment structures—SS-AZ and SS-NJ, for example—with separate parent containers for zones under each deployment structures. Users in each region can select the appropriate parent container when they create new zones.

   > Users must have permission to read and create container objects on the parent Zones container and all child objects. You should verify the appropriate users have the permissions required to create new zones.

10. If you are using the recommended deployment structure, click **Next** to continue.

    This option allows "self-service" join operations for computers in the Computers container. It is only applicable if you are not using the recommended deployment structure. If you want to support "self-service" join operations and are not using the recommended deployment structure, select **Grant computer accounts in the Computers container permission to update their own account information**, then click **Next**.

11. If you plan to use Access Manager to manage information stored in Active Directory and maintain data integrity, click **Next** to continue.

    You should select **Register administrative notification handler for Microsoft Active Directory Users and Computers snap-in** if you want to automatically maintain the integrity of the information in Server Suite profiles.

This option prevents Server Suite profile information from being left “orphaned” when changes are made to Active Directory objects such as users and groups. This option is not selected by default because it requires you to have Enterprise Admin or Domain Admin rights for the forest root domain.

12. Select **Activate Centrify profile property pages** if you want to be able to display Server Suite profiles in any Active Directory context, then click **Next**.

    Setting this option ensures that displaying the properties for a user, group, or computer always displays the Centrify Profile tab regardless of how you navigate to the Properties dialog box.

13. Review and confirm your configuration settings, click **Next**, then click **Finish**.

Installing Agents on Computers to be Managed

This chapter describes the recommended steps for deploying Server Suite software on the nonWindows computers that you want to add to Active Directory. The chapter also describes the alternatives you can use to install agent packages on non-Windows computers, including using native Linux installers to install Server Suite packages manually and automatically.

About the Deployment Process

The steps in this section, and in Preparing to migrate existing users and groups and Migrating existing users to hierarchical zones, are iterative in nature. In most cases, you will select a subset of computers for deployment, and repeat the steps for each target group until you have migrated all of the computers and users in the enterprise into Active Directory.

There is no technical requirement that you only work with a subset of computers at a time, but in practice the process of checking computers for potential problems and resolving open issues is more manageable when applied to a subset of computers. It is also more practical to migrate user populations in stages rather than all at once. After you step through the process a few times, you'll be able to anticipate and resolve potential issues more quickly and move into a more rapid deployment model.

Select a Target Set of Computers

As a first step in preparing to install Server Suite software, you should select a target set of computers on which to deploy. The target set can be based on any criteria you choose. In many organizations, new software must always be installed in the development environment first, then in the pre-production environment, before it can be deployed in the production environment. If your organization has this type of requirement, the first target set of computers would be the computers in the development environment.

Other possible candidates for the target set might be computers that:

- Have been identified for changes by an audit finding
- Are in the same physical location, such as a particular data center
- Share common attributes, such as all Red Hat Linux computers or all of the servers in a Web farm
- Are used by a particular department, project, or line of business
- Have a common set of users who need access to the computer resources

After you have identified a target set of computers, you are ready to begin the deployment. You should notify the user community that you are planning to install software on the target set of computers. For example, you may want

to notify users by sending out an email message similar to the sample provided in Preliminary software delivery notification email template.

After you have identified a target set of computers to work with, you can use adcheck to check whether those computers have any issues that need to be resolved before you install new software on them. Checking the environment before you install helps to reduce change control issues.

Options for deploying Server Suite Agent Packages

You can:

- Run the agent installation script locally on any computer and respond to the prompts displayed.
- Create a configuration file and run the installation script remotely on any computer in silent mode.
- Use the install or update operations in the native package installer for your operating environment.
- Use a commercial or custom software distribution tool.

If you want to use one of these installation options and need more information, see the appropriate section.

Install Interactively on a Computer

The Server Suite Agent installation script, install.sh, automatically checks the operating system, disk space, DNS resolution, network connectivity, and other requirements on a target computer before installing. You can run this script interactively on any supported UNIX, Linux, or Mac OS X computer and respond to the prompts displayed.

To install Server Suite software packages on a computer interactively

1. Log on or switch to the root user if you are installing on a Linux or UNIX.

   If you are installing on Mac OS X, you can log on with any valid user account.

   > On Mac OS X computers, you can install interactively using the graphical package installer or the install.sh script. For information about installing and joining an ActiveDirectory domain using the Mac OS X package installer, see the Mac-specific instructions in the *Administrator's Guide for Mac*.

2. Mount the cdrom device using the appropriate command for the local computer's operating environment, if necessary. On most platforms, the CD drive is automatically mounted.

   > If you have downloaded the package from an FTP server or website, verify the location and go on to the next step.

   The instructions for mounting the CD drive are platform-specific. For example on Linux, you can use a command similar to the following:

   ```
   mount /mnt/cdrom
   ```

   To manually mount the CD drive on AIX, run a command similar to the following:

   ```
   mount -v cdrfs -o ro /dev/cd0 /cdrom
   ```

   To mount the CD drive on HP-UX, run a command similar to the following to display the long file names:

   ```
   mount -F cdfs -o rr /dev/dsk/c0t0d0 /mnt/cdrom
   ```

3. Change to the appropriate directory that contains the Server Suite Agent package you want to install.

   For example, to install an agent on a Linux computer from a downloaded Server Suite ISO or ZIP file, change to the Agent_Linux directory:

   ```
   cd Agent_Linux
   ```

   Similarly, if you are installing on a Solaris, HP-UX, AIX or other UNIX computer, change to the Agent_Unix directory. If installing on a Mac OS X computer, change to the Agent_Mac directory.

   If you downloaded individual agent packages from the Delinea Download Center, unzip and extract the contents. For example:

   ```
   gunzip -d centrify-infrastructure-services-VERSION-platform-arch.tgz
   tar -xf centrify-infrastructure-services-VERSION-platform-arch.tar
   ```

4. Run the install.sh script to start the installation of the agent on the local computer's operating environment. For example:

   ```
   ./install.sh
   ```

5. Follow the prompts displayed to select the services you want to install and the tasks you want to perform. For example, you can choose whether you want to:

   - Perform a default installation.
   - Perform a custom installation by selecting the specific packages to install.
   - Join a domain automatically at the conclusion of the installation.

     Depending on your selections, you may need to provide additional information, such as the user name and password for joining the domain.

Run the Bundle Installation from a Mounted Network Volume

You can install agents from a mounted network volume using the install-bundle.sh script. This script is available on the agent CD or ISO file that contains all of the supported agent platforms in compressed format. The bundle installation script automatically determines the platform required and extracts the contents of the appropriate TGZ file, then starts the normal installation process.

To use the install-bundle.sh script

1. Copy the install-bundle.sh script onto a network file system share and mount the shared directory.
2. Verify that the file is executable and that you have appropriate privileges to run it. For example:

   ```
   chmod +x install-bundle.sh
   chmod 755 install-bundle.sh
   ```

3. Run the script without command line options to start the installation or add command line options to install the agent silently.

   For example, to start an interactive installation, type a command similar to this:

   ```
   sudo ./install-bundle.sh
   ```

   To install the agent silently, type a command similar to this:

   ```
   ./install-bundle.sh --std-suite --adjoin_opt="sidebet.org --password pa\$swd sudo
   ./install-bundle.sh
   ```

```
zone global --container sidebet.org/UNIX/Servers --server demo.sidebet.org"
```

To see complete usage information for the install-bundle.sh script, type:

```
./install-bundle.sh --help
```

Install Silently Using a Configuration File

Installing without user interaction enables you to automate software delivery and the management of remote computers. If you want to install files without any user interaction, you can run the install.sh script silently invoking the script with the appropriate command-line arguments. You can also customize the packages installed and other options by creating a custom configuration file for the installer to use.

- To see the install.sh silent mode and other command line options, enter `install.sh -h`
- To install Authentication & Privilege default packages and configuration options silently, run:
  `install.sh --std-suite`
- To install Authentication & Privilege and Audit & Monitoring default packages and configuration options, run:
  `install.sh --ent-suite`
- To install a customized set of packages that all have the same version number, run:
  `install.sh -n`

About the Sample Configuration Files Available

You can customize the install.sh execution script. There are two sample configuration files for installing software packages silently. These sample configuration files are located in the same directory as the install.sh script:

`centrify-suite.cfg`

`centrifydc-install.cfg`

If you want to customize the packages installed or other configuration options, you can modify the sample `centrify-suite.cfg` or `centrifydc-install.cfg` file.

The `centrify-suite.cfg` file is used when you run install.sh with the `--std-suite` or `--ent-suite` options. If you run `install.sh --std-suite` or `install.sh --ent-suite` with a customized version of the `centrify-suite.cfg` file, you can selectively install compatible add-on packages that do not have the same version number as the core Server Suite Agent.

Alternatively, you can run `install.sh -n` with a customized version of the `centrifydc-install.cfg` file to install the agent and add-on packages if they all have the same version number.

If you run the install.sh script silently and it cannot locate the `centrify-suite.cfg` or `centrifydc-install.cfg` file to use, default values defined directly in the script itself are used.

Setting the Parameters in a Custom Configuration File for the Installation Script

If you want to specify values for the install.sh script to use, you should edit the sample `centrify-suite.cfg` or `centrifydc-install.cfg` file in its default location before invoking the install.sh script in silent mode.

The parameters in the `centrifydc-install.cfg` or `centrify-suite.cfg` file are the same, except that the `centrify-suite.cfg` file is used when installing a set of services to allow packages with different version numbers to be installed together. Because you should not modify the compatibility defined in the `centrify-suite.cfg` file, those parameters are not included in the table.

To customize the installation using the `centrifydc-install.cfg` or `centrify-suite.cfg` file, you can set the following parameters:

Parameter	Description
ADCHECK	Indicate whether you want to run the `adcheck` program to check the configuration of a local computer and its connectivity to Active Directory. Note that the install.sh script calls `adcheck` twice. After the first call, `adcheck` performs several required pre-installation steps to make sure you can install the Centrify Agent on the host computer. These steps are mandatory and cannot be skipped. However, the second call to `adcheck` is used to perform post-installation steps to make sure the agent has been installed successfully. The second set of checks is optional and can skipped. Set this parameter to Y if you want to run `adcheck` after installing. For non-interactive installations, the default is N.
ADLICENSE	Indicate whether you want to install licensed features. Set this parameter to Y if you have purchased and installed license keys. If you downloaded and want to install unlicensed Server Suite Free agents, set this parameter to N.
GLOBAL_ ZONE_ONLY	Specify whether you want to install the agent in a Solaris 10 global zone and no other zones. Set this parameter to Y only if you are running the install.sh script on a Solaris 10 computer and want to install the agent in the Solaris 10 global zone and none of your non-global zones. In most cases, you only set this parameter to Y if you use sparse root zones. The default setting for this parameter is N so that the agent is installed in all Solaris zones. If the script is not running on a Solaris 10 computer, this parameter is ignored.
ADJOIN	Indicate whether you want to attempt to join an Active Directory domain in non-interactive mode. Set this parameter to Y to attempt to join the domain automatically. Set this parameter to N to manually join the domain after installation.
ADJ_FORCE	Overwrite the information stored in Active Directory for an existing computer account. Set this parameter to Y to replace the information for a computer previously joined to the domain. If there is already a computer account with the same name stored in Active Directory, you must use this option if you want to replace the stored information. You should only use this option when you know it is safe to force information from the local computer to overwrite existing information.
ADJ_TRUST	Set the **Trust for delegation** option in Active Directory for the computer account. Trusting an account for delegation allows the account to perform operations on behalf of other accounts on the network.

DOMAIN	Specify the domain to join, if you set the ADJOIN parameter to Y. Set this parameter to the name of a valid Active Directory domain.
USERID	Specify the Active Directory user name to use when connecting to Active Directory to join the domain. Set this parameter to a valid Active Directory user name.
PASSWD	Specify the password for the Active Directory user name you are using to connect to Active Directory. Set this parameter to the password for the Active Directory user name specified for the USERID parameter.
COMPUTER	Specify the computer name to use for the local host in Active Directory. Set this parameter to the computer name you want to use in Active Directory if you don't want to use the default host name for the computer.
CONTAINER	Specify the distinguished name (DN) of the container or Organizational Unit in which you want to place this computer account. The DN you specify does not need to include the domain suffix. The domain suffix is appended programmatically to provide the complete distinguished name for the object. If you do not specify a container, the computer account is created in the domain's default Computers container. Note that the container you specify must already exist in Active Directory, and you must have permission to add entries to the specified container.
ZONE	Specify the zone to which you want to add this computer.
SERVER	Specify the name of the domain controller to which you prefer to connect. You can use this option to override the automatic selection of a domain controller based on the Active Directory site information.
DA_ENABLE	Indicate whether you want to automatically enable the auditing service on the local computer. The valid settings are: Y if you want to enable auditing with the default auditing configuration. N if you don't want to enable auditing. K if you are upgrading and want to keep your current auditing configuration unchanged.
DA_X_ ENABLE	Indicate whether you want to automatically enable the Linux desktop auditing service on the local computer. The valid settings are: Y if you want to desktop enable auditing with the default auditing configuration. N if you don't want to enable desktop auditing. K if you are upgrading and want to keep your current auditing configuration unchanged
DA_INST_ NAME	Specify the name of an auditing installation if you set the DA_ENABLE parameter to Y.
REBOOT	Indicate whether you want to automatically restart the local computer after a successful installation. Set this parameter to Y if you want to automatically restart the local computer or to N if you don't want the computer restarted automatically.
INSTALL	
UNINSTALL	Specify whether you want to forcibly uninstall all installed packages.

```
ADCHECK="N"
ADLICENSE="Y"
# Solaris 10 -G option, installation in global zone only
GLOBAL_ZONE_ONLY="N"
ADJOIN="Y"
ADJ_FORCE="N"
ADJ_TRUST="N"
DOMAIN="sample.company.com"
USERID=administrator
PASSWD="securepassword123"
# COMPUTER=my_host_name
# CONTAINER="my_computers"
ZONE="global_zone"
# SERVER=server_name
DA_ENABLE="N"
DA_INST_NAME=""
REBOOT="Y"
# Install the core agent package
INSTALL="Y"

# Skip installation for other packages
CentrifyDC_nis=
CentrifyDC_krb5=
CentrifyDC_ldapproxy=
CentrifyDC_openssh=
CentrifyDC_web=
CentrifyDC_apache=
CentrifyDC_idmap=
CentrifyDA=
```

This sample configuration file does not install any of the Server Suite add-on packages. You can also use the configuration file to silently install or update selected packages. For example, to update the LDAP proxy service and OpenSSH on a computer, you would modify the configuration file to indicate that you want to update those packages:

```
CentrifyDC_ldapproxy="U"
CentrifyDC_openssh="U"
```

Customizing the Return Codes for the Installation Script

Normally, when you run the install.sh script silently, the script returns an exit code of 0 if the operation is successful. If you want the script to return exit codes that indicate whether the operation performed was a successful new installation, a successful upgrade, a successful uninstall, or there were errors preventing installation, you can also use the custom_rc option. For example:

```
install.sh -n --custom_rc
```

When you specify this option, the following return codes that are defined in the install.sh script are used to provide more detailed information about the result:

Return code	Description
CODE_SIN=0	Successful installation

CODE_SUP=0	Successful upgrade
CODE_SUN=0	Successful uninstallation
CODE_NIN=24	Did nothing during installation
CODE_NUN=25	Did nothing during uninstallation
CODE_EIN=26	Error during installation
CODE_EUP=27	Error during upgrade
CODE_EUN=28	Error during uninstallation
CODE_ESU=29	Error encountered during setup, for example, the UID is not the root user UID, the operating environment is not supported or not recognized, or the script is executed with invalid arguments

Use Other Automated Software Distribution Utilities

You can also install Server Suite software using virtually any automated software distribution framework. For example, you can use software delivery offerings from HP OpsWare or IBM Tivoli, or features such as Apple Remote Desktop, or software distribution in the Casper Suite to deliver Server Suite software to remote computers. You can also use any custom software delivery tools you have developed specifically for your organization. If you use a commercial or custom software distribution mechanism, review the release notes text file included with agent package for platform-specific installation details.

About the Files And Directories Installed on the Agent

When you complete the installation, the local computer will be updated with the following directories and files for the core Server Suite Agent for *NIX:

This directory	Contains
`/etc/centrifydc`	The agent configuration file and the Kerberos configuration file.
`/usr/share/centrifydc`	Kerberos-related files and service library files used by the Centrify Agent to enable group policy and authentication and authorization services.

`/usr/sbin /usr/bin`	Command line programs to perform Active Directory tasks, such as join the domain and change a user password.
`/var/centrify`	Directories for temporary and common files that can be used by the agent.
`/var/centrifydc`	Before joining the domain, the directory contains basic information about the environment, such as the IP address of the DNS server and whether you installed licensed or express agent features. After you join the domain, several files are added to this directory to record information about the Active Directory domain the computer is joined to, the Active Directory site the computer is part of, and other details.

Depending on the components you select during installation, additional files and directories might be installed or updated. For example, if you install Enterprise Edition, the computer is updated with additional files and directories for auditing.

Joining an Active Directory Domain at a Later Time

At this point, you have delivered the software to target computers, but not changed their configuration. Users still have exactly the same access as they did before installing Server Suite software. The computer's configuration changes only happen when the computer joins an Active Directory domain, that is, joining the domain is what "activates" Server Suite software.

You have the option to automatically join an Active Directory domain when you install Server Suite Agents the install.sh script. In most cases, however, you should not do so unless you have already planned your user migration and created your initial zones. Typically, it is best to analyze the user population and prepare for migration before joining the domain to ensure minimal disruption of user activity and ease the transition to new software. Over time, as you become more familiar with the migration process and refine your zone design, you can adapt the steps to suit your organization.

If you want to join the domain at the same time you deploy the Server Suite software, you should do the following before you install files on the UNIX computers:

1. Download the Server Suite software for all platforms or the subset of platforms you intend to support.
2. Analyze existing user and group accounts.
3. Identify your zone requirements and create the initial zone design.
4. Migrate users and groups into the appropriate zones and role assignments.
5. Use the install.sh script or a custom script to install Server Suite Agents and join the domain.

The additional steps are described in the next sections. You can also manually join a domain at any time after installation by using the `adjoin` command.

Installing the Agent on Solaris Systems

This section covers information about installing the Server Suite Agent for *NIX on Solaris systems. The procedures differ depending on whether you're installing svr4 or IPS packages. If you're installing IPS packages onto a system with Solaris 11 child zones, there's a separate procedure for that deployment.

For which packaged file to use for installation, refer to the table below.

Solaris version	Package type	x86 or Sparc	Agent package filename
Solaris 10	svr4	x86	centrify-server-suite-2021.1-sol10-x86.tgz
Solaris 10	svr4	Sparc	centrify-server-suite-2021.1-sol10-sparc.tgz
Solaris 11	svr4	x86	centrify-server-suite-2021.1-sol10-x86.tgz
Solaris 11	svr4	Sparc	centrify-server-suite-2021.1-sol10-sparc.tgz
Solaris 11	IPS	x86	centrify-server-suite-2021.1-sol11-i386.tgz
Solaris 11	IPS	Sparc	centrify-server-suite-2021.1-sol11-sparc.tgz

Installing the Solaris Svr4 Agent Packages

Download the solaris package appropriate for your Solaris system and run the install.sh script as mentioned in the "Install interactively on a computer" section. You can follow the same procedure if you're installing on a system with or without child zones.

You can run the following command to verify the Solaris agent package svr4 installation status:

```
pkginfo | grep -i centrify
```

Note that there is no space between "pkg" and "info"; if you search for "pkg info" you'll be searching for IPS packages.

Installing the Solaris IPS Agent Packages

This procedure is for systems where you are doing a fresh install of Server Suite software onto a Solaris 11 system with IPS support.

This procedure is the same as for the regular install script, except that you run the `install-ips.sh` script, not the install.sh script -- see the "Install interactively on a computer" section.

You'll need the centrify-infrastructure-services-VERSION-sol11-i386.tgz file or the `centrify-infrastructure-services-VERSION-sol11-sparc.tgz` file for this procedure, depending on the type of system you have.

To install the Solaris IPS packages

1. Download the Server Suite package for your version of Solaris.
2. Extract the Server Suite packages onto the system.
3. Run the `install-ips.sh` script. For example:

   ```
   ./install-ips.sh
   ```
4. Follow the prompts displayed to select the services you want to install and the tasks you want to perform. For example, you can choose whether you want to:
   - Perform a default installation.
   - Perform a custom installation by selecting the specific packages to install.

- Join a domain automatically at the conclusion of the installation.

  Depending on your selections, you may need to provide additional information, such as the user name and password for joining the domain.

5. You can run the following command to verify the Solaris agent package IPS installation status:

   `pkg info | grep -i centrify`

   Note the space between "pkg" and "info"; if you search for "pkginfo" you'll be searching for svr4 packages.

Installing the Solaris IPS Agent Packages With Child Zones

When you install the agent onto a Solaris 11 computer enabled with IPS that also has one or more child zones configured, you need to import the agent packages into a new repository and then install directly from the repository.

You do this install in the global zone and the repository will automatically install the files into the child zones.

You'll need the `centrify-infrastructure-services-VERSION-sol11-i386.tgz` file or the `centrify-infrastructure-services-VERSION-sol11-sparc.tgz` file for this procedure, depending on the type of system you have.

To install the Solaris IPS agent packages onto a system with one or more child zones

1. Create a directory and extract the IPS tgz file into that directory.

   For example, create directory called "install-ips" and extract the contents of the tgz file into that directory.

   a. Create a repository:

      For example, run the following command to create a repository called "my-repo":

      Tip: You can run the zfs list command to list all of the zone file systems.

      `zfs create rpool/export/my-repo`

      zfs set atime=off rpool/export/my-repo

      pkgrepo create /export/my-repo

      pkgrepo set -s /export/my-repo publisher/prefix=centrify

      pkgrepo -s /export/my-repo refresh

      pkgrepo -s /export/my-repo info

      pkg set-publisher -G '*' -M '*' -g /export/my-repo centrify

      pkg publisher

      You should see the repository listed.

      `PUBLISHER TYPE STATUS P LOCATION`

      centrify origin online F file:///export/my-repo

2. Import the packages into the repository. You need to import the packages that end with .p5p and you need to import them one at a time.

a. In the directory where you extracted the Server Suite Agent packages, list out the files in that directory (use the ls command).

   The full package list of files that you need to import into the repository looks something like this:

   ```
   centrifyda-3.7.0-sol11-i386.p5p
   ```

   centrifydc-5.7.0-sol11-i386.p55

   centrifydc-curl-5.7.0-sol11-i386.p5p

   centrifydc-ldapproxy-5.7.0-sol11-i386.p5p

   centrifydc-nis-5.7.0-sol11-i386.p5p

   centrifydc-openldap-5.7.0-sol11-i386.p5p

   centrifydc-openssh-5.7.0-sol11-i386.p5p

   centrifydc-openssl-5.7.0-sol11-i386.p5p

b. Import each package into the repository.

   For example, if you're installing all 8 packages, you'll run the following 8 commands:

   ```
   pkgrecv -s centrifyda-3.7.0-sol11-i386.p5p -d /export/my-repo '*'
   ```

   pkgrecv -s centrifydc-5.7.0-sol11-i386.p55 -d /export/my-repo '*'

   pkgrecv -s centrifydc-curl-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

   pkgrecv -s centrifydc-ldapproxy-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

   pkgrecv -s centrifydc-nis-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

   pkgrecv -s centrifydc-openldap-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

   pkgrecv -s centrifydc-openssh-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

   pkgrecv -s centrifydc-openssl-5.7.0-sol11-i386.p5p -d /export/my-repo '*'

c. You can verify that the packages imported correctly by listing out the repository packages.

   For example, run the following command:

   ```
   pkgrepo list -s /export/my-repo
   ```

   You'll see a list of packages where each package has a long version. For example:

   ```
   centrify security/centrifydc 5.7.0.207:20200726T052946Z
   ```

3. Install the packages from the repository into the parent and child zones.

   Be sure to reference the package's entire version. When you install the `centrifydc` package, the other packages for cURL, OpenLDAP, and OpenSSL are also installed.

   For example, to install centrifydc, you'd run the following command, :

   ```
   pkg install -r security/centrifydc@5.7.0.207:20200726T052946Z
   ```

   For example, to install all the packages with one command, you'd run something like this:

   ```
   pkg install -r security/centrifydc@5.7.0.207:20200726T052946Z security/centrifydc-
   ldapproxy@5.7.0.207:20200726T053320Z security/centrifydc-nis@5.7.0.207:20200726T053352Z
   ```

```
security/centrifydc-openssh@5.7.0.207:20200727T065442Z
security/centrifyda@3.7.0.171:20200725T014652Z
```

4. You can run the following command to verify the Solaris agent package IPS installation status:

   ```
   pkg info | grep -i centrify
   ```

   Note the space between "pkg" and "info"; if you search for "pkginfo" you'll be searching for svr4 packages.

Uninstalling the Agent on Solaris Systems

To uninstall the Solaris svr4 packages

1. In the directory where you have downloaded and extracted the Centrify Agent packages, run the following command:

   ```
   ./install.sh -e -n
   ```

2. You can run the following command to verify the Solaris agent package svr4 installation status:

   ```
   pkginfo | grep -i centrify
   ```

   Note that there is no space between "pkg" and "info"; if you search for "pkg info" you'll be searching for IPS packages.

To uninstall the Solaris IPS packages

1. In the directory where you have downloaded and extracted the Server Suite Agent packages, run the following command:

   ```
   ./install-ips.sh -e -n
   ```

2. You can run the following command to verify the Solaris agent package IPS installation status:

   ```
   pkg info | grep -i centrify
   ```

   Note the space between "pkg" and "info"; if you search for "pkginfo" you'll be searching for svr4 packages.

To uninstall the Solaris IPS packages on systems with one or more child zones

1. To uninstall a single package from both parent and child zones, run the following command:

   ```
   pkg uninstall -r packagename
   ```

   For example, to uninstall only the CentrifyDA package, run the following command:

   ```
   pkg uninstall -r security/centrifyda
   ```

   To uninstall more than one package or all installed packages, enter the package names separated by a space. For example:

   ```
   pkg uninstall -r security/centrifyda security/centrifydc-curl security/centrifydc-ldapproxy
   ```

2. You can run the following command to verify the Solaris agent package IPS installation status:

   ```
   pkg info | grep -i centrify
   ```

   Note the space between "pkg" and "info"; if you search for "pkginfo" you'll be searching for svr4 packages.

Sun Solaris Installation Notes

This section describes the unique characteristics or known limitations that are specific to using authentication service on a computer with the Solaris operating environment.

Changing the Local User Password on Solaris

On Solaris, the passwd command is designed to update the databases listed in the `nsswitch.conf` file or the specific repositories you indicate with the -r option. Therefore, by default, you can use passwd command without any command line options to update your password wherever necessary.

Once you install authentication service and join the domain, however, Active Directory becomes the primary repository for user account information and changing the password for any local user account you need to maintain outside of Active Directory requires you to explicitly specify the repository to update with the -r option.

For example, if you want to change the password for a local user account in`/etc/passwd`, you must specify the files repository when you run the passwd command:

```
passwd -r files user
```

If you want to update the password for an Active Directory user account, you can use the `passwd` command without the repository option on Solaris 10. For example:

```
passwd adusername
```

If you are using an earlier version of the Solaris operating environment, however, you must use the `adpasswd` command that is installed with authentication service to update the password for Active Directory user accounts. For information about using `adpasswd`, see the `adpasswd` man page or the Administrator's Guide for Linux and UNIX.

Installing Authentication Service Packages into Solaris 10 Zones

All zones should be up and running during an upgrade from a previous release of Server Suite Authentication Service and its add-on packages (for example sudo or Server Suite for Web Applications) should not be installed directly into a sparse zone, they should be installed from the global zone only.

Installing Authentication Service Packages into Solaris 11 Child Zones

You need to install SVR4 packaging tools in the child zone before authentication service can be installed.

To check if the SVR4 package has been installed, run

```
$ pkg info svr4
```

If it is not installed yet, run the following to install it:

```
$ pkg install pkg:/package/svr4
```

Note that the command above may need internet connection (depends on how the IPS repository is configured in the zone).

Creating a Home Directory for New Users on Solaris

In most operating environments, when new users log on successfully, the authentication service will automatically create the user's home directory. On Solaris, however, the home directory is typically auto-mounted over NFS, so

the option to automatically create a new home directory for new users is off by default. You can turn on this feature, if suitable to your environment, by adding the following to `/etc/centrifydc/centrifydc.conf`:

```
pam.create.homedir: true
```

With this flag, the first time a user logs in the home directory will be created. The user will see the message "Failed to create home directory", but this can be ignored.

In Express mode use `auto.schema.homedir` to specify the home directory for users. Use %{user} as a placeholder for a user's name.

For example:

```
auto.schema.homedir: /export/home/${user}
```

Using a Native Package Installer

If you want to manually install a software package using a native installation program instead of the Server Suite installation script, you can follow the instructions in the *Upgrade and Compatibility Guide* for the most common native package installers, such as the Red Hat or Debian package manager. You should note that these native packages are signed with a GNU Privacy Guard (GPG) key. You need to import the key to verify the package authenticity before installing the package. You can download the `RPM-GPG-KEY-centrify` file from the Delinea Download Center.

Alternatively, you can use any other installation program you have available for the local operating environment. For example, if you use another program such as SMIT, YAST, APT, SUSE, or YUM to install and manage software packages, you can use that program to install Server Suite software packages.

Perform the following steps to install the Server Suite Agent using a native installation program that does not require a connection to a package repository. To use a native installation program that requires a repository connection (such as yum, SUSE or APT), see Enabling package repositories.

To install the agent using a native installation program

1. Log on as or switch to the root user.
2. If you are installing from a CD and the CD drive is not mounted automatically, use the appropriate command for the local computer's operating environment to mount the cdrom device.
3. Copy the appropriate package for the local computer's operating environment to a local directory.

   For example, if installing from the CD and the operating environment is Solaris 10 SPARC:

   `cp /cdrom/cdrom0/Unix/centrifydc-`**release**`-sol10-sparc-local.tgz`.
4. If the software package is a compressed file, unzip and extract the contents. For example, on Solaris:

   ```
   gunzip -d centrifydc-release-sol10-local.tgztar -xf centrifydc-release-sol10-sparc-local.tar
   ```
5. Run the appropriate command for installing the package based on the local computer's operating environment. For example, on Solaris:

   ```
   pkgadd -d CentrifyDC-a admin
   ```

   If you are not sure which command to use for the local operating environment, see the documentation associated with the package installer you are using.

Enabling Package Repositories

You can also download and install agents using Linux package management software for your operating system. To do this, you set up a repository for your operating system and then use the software's command line tools to manage automatic agent updates.

- **RedHat, CentOS, or Amazon systems**: Use the "Yellowdog Updater, Modified (yum)" tool to update the rpm-redhat repository. For details, seeTo set up and configure a RedHat, CentOS, or Amazon repository.
- **SuSE systems**: Use the Zypper tool to update the rpm-suse repository. For details, see To set up and configure a SUSE repository.
- **Debian or Ubuntu systems**: Use the Advanced Package Tool (APT), apt-get, or Aptitude tools to update the deb repository. For details, seeTo set up and configure a Debian or Ubuntu repository.
- **Atomic systems**: Use curl or wget to update the wget repository. For details, see To access a raw package (WGET) repository.
- **Alpine Linux systems**: For details, see To set up and configure an Alpine Linux repository.

You must perform one of the above procedures to enable the repository.

> The procedures in this section require that you log in to the Delinea Support Portal and go to the Delinea Repo site. On that page, click the link to generate the repo key. You will then specify the repo key in a yum (RHEL, SUSE, and so forth) or APT (Debian, Ubuntu, and so forth) configuration file. There are some examples on the Delinea repo site about how to add the key to your configuration file.

> For additional details about configuring and using SUSE or yum repositories, see the documentation for the distribution of Linux you are using. For additional details about configuring and using APT repositories, see the documentation for the distribution of Debian Linux or Ubuntu you are using.

WARNING: If you specify your repository on the command line, be sure to clean out your command history afterwards. Because the URL for your repository includes the credentials to access it, leaving this information around in command history is not a secure practice.

To Set Up and Configure an Alpine Linux Repository

1. To configure the repository automatically, run the following commands:

   ```
   sudo apk add --no-cache bash
   ```

   curl -1sLf \ 'https://cloudrepo.centrify.com/URLTOKEN/apk/setup.alpine.sh' \ | sudo -E bash

   Note: Enter your Delinea repository URL token in place of `URLTOKEN`.

2. Or, if you want to manually configure the repository, run the following commands:

   ```
   curl -1sLf 'https://cloudrepo.centrify.com/URLTOKEN/apk/rsa.5DD8742729E6E4B2.key' >
   /etc/apk/keys/apk@centrify-5DD8742729E6E4B2.rsa.pub
   ```

   curl -1sLf 'https://cloudrepo.centrify.com/URLTOKEN/apk/config.alpine.txt?distro=alpine&codename=v3.13' >> /etc/apk/repositories

   apk update

When configuring the repository manually, you reference the Delinea public RSA key: `apk@centrify-5DD8742729E6E4B2.rsa.pub`.

1. Execute the apk add command to install the Server Suite packages. For example:

   ```
   # apk add centrifydc centrifydc-nis
   ```

   To uninstall the Server Suite Agent for *NIX rpm, you can use the `del` command to delete the Server Suite Agent for *NIX package. For example:

   ```
   # apk del centrifydc=5.8.0-xxx
   ```

To Access a Raw Package (wget) Repository for Atomic

Use the WGET repository for Atomic packages or any raw or plain files such as *.zip, *.tar, *.tgz, and so forth .

1. Browse the package index to determine which file you want to download. You can view the package index in HTML or JSON:
   - HTML version is at `https://cloudrepo.centrify.com/URLTOKEN/wget/raw/`
   - JSON version is at `https://cloudrepo.centrify.com/URLTOKEN/wget/raw/index.json`

     Note: Enter your Centrify repository URL token in place of `URLTOKEN`.
2. Download the desired file using either `curl` or `wget`. For example:

   ```
   curl -1 -O 'https://cloudrepo.centrify.com/URLTOKEN/wget/raw/versions/Latest/CentrifyDC-atomic.x86_64.tgz'
   ```

   or

   ```
   wget 'https://cloudrepo.centrify.com/URLTOKEN/wget/raw/versions/Latest/CentrifyDC-atomic.x86_64.tgz'
   ```

Debian Ubuntu

To set up and configure a Debian or Ubuntu repository

1. Create the repository:

   You can manually create the repository or you can use a setup script to create the repository automatically.
   - **_To create the Debian or Ubuntu repository configuration file manually**
     a. Update the `/etc/apt/sources.list` file to include the official Delinea package repository.

        ```
        deb https://cloudrepo.centrify.com/URLTOKEN/deb/deb/debian any-version main
        /etc/apt/sources.list.d/centrify-deb.list
        ```
2. Import your GPG key and update the repository.

   ```
   # bash -c 'wget -O - https://support.delinea.com/s/product-downloads/products/RPM-GPG-KEY-centrify | apt-key add -'
   ```
3. Comment out the `no-debsig` line in `/etc/dpkg/dpkg.cfg` to enable GPG signature validation.

   ```
   # grep no-debsig /etc/dpkg/dpkg.cfg
   ```

   # no-debsig

4. Clean and update the local archives.

   ```
   # apt-get clean
   ```

   # apt-get update

- **To create the Debian or Ubuntu repository configuration file automatically from a script**

  ```
  curl -1sLf 'https://cloudrepo.centrify.com/URLTOKEN/deb/cfg/setup/bash.deb.sh' | sudo -E bash
  ```

Note: Enter your Delinea repository URL token in place of `URLTOKEN`.

If you manually created your APT repository, the configuration details are in the `/etc/apt/sources.list` file. If you used the setup script to create the APT repository, the configuration details are in a separate file such as centrify-deb.list in the `/etc/apt/sources.list.d` directory.

2. Execute the apt list command to verify the repository connection. You should see output similar to the following.

   ```
   # apt list --all-versions | grep centrify
   ```

   centrifyda/buster 3.7.0-172 amd64

   centrifyda/buster 3.6.1-324 amd64

   centrifydc-adbindproxy/buster 5.7.0-217 amd64

   centrifydc-adbindproxy/buster 5.6.1-334 amd64

   centrifydc-cifsidmap/buster 5.7.0-207 amd64

   centrifydc-cifsidmap/buster 5.6.1-330 amd64

   centrifydc-curl/buster 5.7.0-207 amd64

   centrifydc-curl/buster 5.6.1-330 amd64

   centrifydc-ldapproxy/buster 5.7.0-207 amd64

   centrifydc-ldapproxy/buster 5.6.1-330 amd64

   centrifydc-nis/buster 5.7.0-207 amd64

   centrifydc-nis/buster 5.6.1-330 amd64

   centrifydc-openldap/buster 5.7.0-207 amd64

   centrifydc-openldap/buster 5.6.1-330 amd64

   centrifydc-openssh/buster 8.2p1-5.7.0.207 amd64

   centrifydc-openssh/buster 7.9p1-5.6.1.329 amd64

   centrifydc-openssl/buster 5.7.0-207 amd64

   centrifydc-openssl/buster 5.6.1-330 amd64

   centrifydc/buster 5.7.0-207 amd64

   centrifydc/buster 5.6.1-330 amd64

3. Execute the `apt install` or `apt-get install` commands to install Delinea packages. For example:

   ```
   # apt install centrifydc centrifydc-nis
   ```

   # apt-get install centrifydc-5.7.0-207

Note: To uninstall the Server Suite Agent for *NIX rpm, you can use the `remove` command to delete the Server Suite Agent for *NIX package or the `purge` command to also delete any configuration files. For example:

```
# apt remove centrifydc=5.7.0-207
```

Redhat

To Set Up and Configure a Redhat, Centos, or Amazon Repository

1. Create a /etc/yum.repos.d/centrify-rpm-redhat.repo configuration file to use the official Delinea package repository, and download a `RPM-GPG-KEY-centrify` key from the Delinea Support Portal.

   You can manually create the configuration file or you can use a setup script to generate the file automatically.

   Note: For the baseurl parameter, enter your Delinea repo URL token in place of **<URLtoken>**.

   - **To create the repository configuration file manually**

     Create a file with the following:

     ```
     # Source: CENTRIFY
     ```

     # Repository: CENTRIFY / rpm-redhat

     # Description: YUM repository for RedHat packages (RPMs)

     [centrify-rpm-redhat]

     name=centrify-rpm-redhat

     baseurl=https://cloudrepo.centrify.com/URLTOKEN/rpm-redhat/rpm/any-distro/any-version/$basearch

     repo_gpgcheck=1

     enabled=1

     gpgkey=https://downloads.centrify.com/products/RPM-GPG-KEY-centrify

     gpgcheck=1

     sslverify=1

     sslcacert=/etc/pki/tls/certs/ca-bundle.crt

     metadata_expire=300

     pkg_gpgcheck=1

     autorefresh=1

     type=rpm-md

- **To create the repository configuration file automatically from a script**

  ```
  curl -1sLf ' https://cloudrepo.centrify.com/URLTOKEN/rpm-redhat/cfg/setup/bash.rpm.sh' |
  sudo -E bash
  ```

You should see output that lists out your repository details, such as the following example:

```
# Source: CENTRIFY
```

Repository: CENTRIFY / rpm-redhat

Description: YUM repository for RedHat packages (RPMs)

[centrify-rpm-redhat]

name=centrify-rpm-redhat

baseurl=https://cloudrepo.centrify.com/URLTOKEN/rpm-redhat/rpm/el/6/$basearch

repo_gpgcheck=1

enabled=1

gpgkey=https://cloudrepo.centrify.com/URLTOKEN/rpm-redhat/cfg/gpg/gpg.BDD3FD95B65ECA48.key

gpgcheck=1

sslverify=1

sslcacert=/etc/pki/tls/certs/ca-bundle.crt

metadata_expire=300

pkg_gpgcheck=1

autorefresh=1

type=rpm-md

Note: The gpgkey listed in the output is a public key.

2. Execute the `yum info` command to verify the repository connection. You should see output similar to the following.

```
#yum info CentrifyDC
```

Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager

2020-11-24 10:20:22,669 [INFO] yum:17896:MainThread @connection.py:905 - Connection built: host=subscription.rhsm.redhat.com port=443 handler=/subscription auth=identity_cert ca_dir=/etc/rhsm/ca/ insecure=False

2020-11-24 10:20:22,671 [INFO] yum:17896:MainThread @repolib.py:464 - repos updated: Repo updates

Total repo updates: 0

Updated

<NONE>

Added (new)

<NONE>

Deleted

<NONE>

This system is not registered with an entitlement server. You can use subscription-manager to register.

Available Packages

Name : CentrifyDC

Arch : x86_64

Version : 5.7.0

Release : 207

Size : 23 M

Repo : centrify-rpm-redhat/x86_64

Summary : Centrify DirectControl Agent

URL : http://www.centrify.com/

License : BSD with portions copyright (c) Centrify Corporation 2006-2020 and licensed under Centrify End User License Agreement

Description : RPM to install Centrify DirectControl on Linux platforms.

```
```

3. Install the Server Suite Agent for *NIX rpm package.

   ```
   # yum install CentrifyDC
   ```

   Note: To uninstall the Server Suite Agent rpm file, you can use the erase command. For example:

   ```
   # yum erase CentrifyDC
   ```

SuSE

To Set Up and Configure a Suse Repository

1. If you have used a Delinea repository before, it's recommended that you first delete the old repositories in the `/etc/zypp/repos.d/*centrify*` directory.
2. Create a new SUSE repository.

   You can manually create the repository or you can use a setup script to create the repository automatically.

   Note: For the baseurl parameter, enter your Delinea repository URL token in place of `URLTOKEN`.

   - **To create the SUSE repository configuration file manually**

     Create a file with the following:

     ```
     /etc/zypp/repos.d # cat /etc/zypp/repos.d/centrify-rpm-suse.repo
     [centrify-rpm-suse]
     name=centrify-rpm-suse
     enabled=1
     autorefresh=1
     baseurl=https://cloudrepo.centrify.com/URLTOKEN/rpm-suse/rpm/any-distro/any-version/$basearch
     type=rpm-md
     repo_gpcheck=1
     gpgcheck=1
     gpgkey=https://downloads.centrify.com/products/RPM-GPG-KEY-centrify
     ```

   - **To create the SUSE repository configuration file automatically from a script**

     ```
     curl -1sLf 'https://cloudrepo.centrify.com/URLTOKEN/rpm-suse/cfg/setup/bash.rpm.sh' | sudo
     -E bash
     ```

3. Refresh the cache.

   ```
   /etc/zypp/repos.d # zypper refresh
   ```

4. Verify the connection to the repository.

   ```
   /etc/zypp/repos.d # zypper packages |grep centrify
   ```

   You should see output similar to the following:

```
mysusemachine:/etc/zypp/repos.d # zypper refresh
Retrieving repository 'centrify-rpm-suse' metadata ----------[\]
Retrieving repository 'centrify-rpm-suse' metadata .....[done]
Building repository 'centrify-rpm-suse' cache .....[done]
All repositories have been refreshed.
utsles15ppcle:/etc/zypp/repos.d # zypper packages | grep Centrify
| centrify-rpm-suse | CentrifyDA          | 3.7.0-172       | ppc64le
| centrify-rpm-suse | CentrifyDA          | 3.6.1-324       | ppc64le
| centrify-rpm-suse | CentrifyDC          | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC          | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-cifsidmap | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-cifsidmap | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-curl     | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-curl     | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-ldapproxy | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-ldapproxy | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-nis      | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-nis      | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-openldap  | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-openldap  | 5.6.1-330       | ppc64le
| centrify-rpm-suse | CentrifyDC-openssh   | 8.2p1-5.7.0.207 | ppc64le
| centrify-rpm-suse | CentrifyDC-openssh   | 7.9p1-5.6.1.329 | ppc64le
| centrify-rpm-suse | CentrifyDC-openssl   | 5.7.0-207       | ppc64le
| centrify-rpm-suse | CentrifyDC-openssl   | 5.6.1-330       | ppc64le
```

5. Install the Server Suite Agent rpm package.

   ```
   # zypper install CentrifyDC
   ```

   Note: To uninstall the Server Suite Agent rpm file, you can use the remove command. For example:

   ```
   # zypper remove CentrifyDC
   ```

Planning to Use Server Suite Zones

One of the most important aspects of managing computers with Server Suite software is the ability to organize computers, users, and groups into **zones**. This section discusses the primary reasons for using zones and provides an overview of how to analyze and migrate an existing user population into zones, including an introduction to assigning roles that enable users to access computer resources. These topics will then be described in greater

detail in the next sections to help you create an initial zone structure and migrate users and groups for a target set of computers.

Why Use Zones?

A zone is similar to an Active Directory organizational unit (OU) or NIS domain. Zones allow you to organize the computers in your organization in meaningful ways to simplify the transition to Active Directory and the migration of user and group information from existing identity stores. The primary benefits to using zones are:

- Identity management through user and group profile definitions
- Access and authorization control through rights and role definitions
- Delegated computer management for zone-based administrative tasks

Zones also enable you to centrally manage configuration policies for computers and users through group policies, but for most organizations the key considerations for designing a zone structure involve:

- Identity management because zones enable you to migrate from a complex UID space, where a user can have multiple UIDs or different profile attributes on different computers or a single UID might identify different people depending on the computer being used. With zones, you can associate multiple UNIX profiles with a user and identify the correct profile attributes for any user on any given computer.
- Access and authorization management because zones enable you to grant specific rights to users in specific roles on specific computers. By assigning roles, you can control who has access to which computers.
- Delegated computer management because zones enable you to assign specific administrative tasks to specific users or groups on a zone-by-zone basis, allowing you to establish an appropriate separation of duties. With zones, administrators can be given the authority to manage a given set of computers and users without granting them permission to perform actions on computers in other zones or access to other Active Directory objects.

In most organizations, the first goal in designing the zone structure is to migrate users and computers from an existing identity store, such as NIS, NIS+, local files, or LDAP, to Active Directory and to do so with the least possible disruption to user activity, business services, and the existing infrastructure. Over time, you can also use zones to organize computers along departmental, geographical, or functional lines using whatever strategy works best for your organization.

Although Server Suite supports a **workstation mode** that does not require you to create and manage zones—a single Auto Zone is defined instead—most organizations find using zones to be an essential part of their migration to Active Directory. The next sections provide more information about why zones are an important part of the planning process. For more information about using Auto Zone, see Deploying to a single Auto Zone.

Identity Management Using Zones

For most organizations, it is impractical to attempt to rationalize user accounts across the enterprise to achieve a single global UID for each user. For example, most organizations have multiple identity stores already in use on their current UNIX platforms. These identity stores may include LDAP directories, NIS or NIS+ domains, and local /etc/passwd and /etc/group configuration files. With these multiple identity stores, it is common for a single user to have a different user name, UID, group memberships, or other attributes defined for different computers.

Zones allow you to import the information from these legacy identity stores without consolidating the multiple profiles that each user may have. For example, a single user might have an account in a UNIX LDAP directory, another defined in a NIS domain, and one or more local /etc/passwd files. Zones enable the profiles from these

different identity stores to map to a single Active Directory user account without changing the user profile defined in each of the legacy directories. By keeping the profiles intact, the user's file ownership and log in permissions are not affected by the migration to Active Directory, making the transition from a legacy system to Active Directory more transparent to end users and less of a management burden for the deployment team.

Role-based Access Control and Zones

As a practical matter, you may choose to use Server Suite zones to ease migration to Active Directory by creating a separate zone for each legacy identity store. However, you can also use zones to group computers by department, by function, or by any other criteria you choose. Using zones in this way gives you a great deal of flexibility in controlling who has access to the UNIX, Linux, and Mac OS X computers in your environment and makes it easier to set up account information for new users based on job function or other criteria.

Through role assignments, zones provide a scope of resources particular users can access, allowing you to define who can do what on which computers. For example, all of the computers in the finance department could be grouped into a single zone called "finance" and the members of that zone could be restricted to finance employees and senior managers, each with specific rights, such as log on to a database, update certain files, or generate reports. This gives you better control over access to systems based on well-defined roles. You can also limit access to certain types of applications, such as database management utilities or web services. For example, you can define specific actions specific users are allowed to perform by assigning them different roles in different zones.

Using Zones to Delegate Administrative Duties

Zones can also be useful for grouping computers that form a natural administrative set or that should be managed by different administrative teams. For example, you may want to group computers that are managed by a local support organization in one zone and computers that are managed by a corporate IT group in another zone.

Using zones, you can then control what different groups of users can do within the zones they have permission to access. For example, you can set up regional zones to provide a separation of duties, authorizing users in San Francisco to manage computers and user profiles in their local office while a team in Barcelona has authority to join computers and manage group profiles for offices located in Spain.

Zones provide a convenient way for you to assign individual administrative responsibilities to specific users or groups based on a set criteria, such as department, geographic location, or functional role.

Deploying to a Single Auto Zone

In most cases, if you are deploying on Linux or UNIX computers and have an existing user population to migrate to Active Directory, you would create a hierarchical zone structure of multiple zones. However, multiple zones are not required for all situations. You can greatly reduce the time required and complexity of your deployment if a single zone suits your organization's needs. For example, if you are deploying on Mac OS X or Windows computers or if you have a mix of computer platforms but do not have an existing user population to migrate, you might benefit from deploying agents using the Auto Zone option.

With Auto Zone, you have a single zone for an entire forest. All of the users and groups you have defined in Active Directory for the forest automatically become valid users and groups on the computers that join the Auto Zone. If the forest has a two-way trust relationship with another forest, all Active Directory users defined in that trusted forest are also automatically valid on computers that join the Auto Zone.

If you simply want to use the Active Directory users and groups you have already defined on the nonWindows computers you manage, you can skip the planning and creation of zones and simply add computers to the Auto

Zone when you join the domain. The UNIX profile attributes that are required to access computers in the Auto Zone are then automatically derived from user attributes in Active Directory or from settings defined in group policies or configuration parameters.

> You cannot use Auto Zone to give automatic access to users and groups in a forest or domain with a one-way trust relationship with another forest or domain.

You can use Auto Zone without enabling any group policies or changing any of the default configuration settings. You can also join a domain through the Auto Zone without installing Access Manager. However, you can use group policies or configuration parameters to specify a subset of Active Directory users or groups as valid Auto Zone users. The settings are then enforced on computers in the Auto Zone.

Using Auto Zone can make sense in small or larger organizations if you are not migrating existing users and groups or maintaining legacy UNIX profile attributes. However, if you use Auto Zone, you cannot use zone-specific features. For computers in the Auto Zone, you cannot configure rights and roles, assign roles to users and groups, or provide different profile attributes on different computers.

For information about joining a domain using Auto Zone, see the man page for adjoin or the *Administrator's Guide for Linux and UNIX*. For information about using group policies or configuration parameters, see the *Group Policy Guide* or *Configuration and Tuning Reference Guide*.

Classic and Hierarchical Zones

If you plan to deploy using zones—which is the most common deployment model—you have to option to create **classic** or **hierarchical** zones. Classic zones provide a simple model for organizing computers and backwards compatibility for organizations with older versions of the Server Suite Agent. Hierarchical zones enable you to establish parent-child zone relationships, allowing profile attributes, rights, and roles to be inherited down the zone hierarchy. Classic zones are peers to each other and do not inherit profile attributes, rights, or roles from each other.

One of the first decisions you need to make in planning your zone structure is whether you will use classic zones, hierarchical zones, or some combination of both.

Should You Use Classic Zones?

Classic zones provide a simple structure for delineating users and groups based on a criteria you choose, such as by region or department. They are most appropriate if you have a well-defined and well-managed UNIX namespace with very few users who require special handling because of multiple profiles or conflicting profile attributes.

Classic zones are simple to manage as long as you only need a few. For example, imagine you have three regional zones with no users in common that are managed independently by their own zone administrators with only one enterprise system administrator who must have a profile in each zone. In that scenario, classic zones provide a simple solution because only one user account, the enterprise system administrator, must have a profile in each zone.

However, classic zones are very limited in complex environments where users need profiles in multiple zones or where there are multiple independently-managed UNIX namespaces to migrate to Active Directory. That is because classic zones do not share data across zone boundaries. The data must be created and managed in each zone independently. By contrast, hierarchical zones support inheritance, enabling you to create parent and child zones that share information as needed. Because classic zones do not support inheritance, you cannot use variables to define profile attributes or any other hierarchical zone features.

For most organizations, classic zones are primarily used to enable a new zone that works with pre-5.0 versions of the Server Suite Agent. If you have an older version of Server Suite software installed and already have some zones deployed in your environment, you can continue to use those zones as-is. After upgrading, you then have the option to create any new zones as classic zones to operate within the legacy zone environment or as hierarchical zones.

If you already have zones deployed, you can convert them to hierarchical zones after you deploy the new version of Server Suite software, if you choose to do so. However, there's no requirement for you to convert existing zones to hierarchical zones.

When Should You Use Hierarchical Zones?

For most organizations, Server Suite recommends that you use hierarchical zones for all new zones that you create. Hierarchical zones provide greater flexibility to inherit profile information, rights and role definitions, and user and group role assignments.

Because hierarchical zones allow you to share or override information at any point in the hierarchy, they also allow you to design a simpler zone structure than classic zones and support an easier deployment model. Typically, a simpler zone structure is easier to manage, but hierarchical zones also allow you to implement a very sophisticated zone structure to address complex access control rules, if you so choose.

How Many Zones Do You Need?

The goal in planning to use zones is to have a fairly small number of zones that organize the computers and users in your organization most effectively. As an example, consider an organization where some UNIX computers are used to host financial applications. Those computers are centrally managed by the IT organization, which follows well-established conventions for issuing user login names, user IDs, and home directories. The same organization has a software development group that includes numerous UNIX workstations that are not centrally managed by the IT organization and computers and accounts are added when needed and managed independently.

Because enterprise-wide conventions are not enforced for the UNIX computers in the software development group, it's possible that the local login names and user IDs may conflict with the names and IDs used on the computers running the financial applications. In addition, users in the software development group may use a different convention for their home directories or prefer different login shells.

Without zones, the IT organization would need to eliminate any duplicate user IDs and verify each login name was unique across all of the computers. By placing the computers running the financial applications in one zone and the computers in the development lab in another zone, the IT organization can avoid the overhead of checking and changing existing account information and can set default zone settings, such as different default home directories or login shells, that are most appropriate to the users in each zone.

There are many different approaches you can take to defining the scope of a zone, including organizing by platform, department, manager, application, geographical location, or how a computer is used. The factors that are most likely to affect your initial zone design, however, will involve migrating user and group profiles, identifying the appropriate access control policies and role assignments, and delegating administrative tasks to the appropriate users and groups. For many organizations, the most important issue during the initial deployment is a successful migration of existing users. Using hierarchical zones with the ability to override attributes simplifies this task, helping to reduce the total number of zones you need to deploy.

A Closer Look at Using Zones in a Hierarchical Model

In older versions of Server Suite software, zones were always parallel with each other and did not share or inherit data except through manual processes. Starting with Infrastructure Services 2012, however, you have the option to use hierarchical zones that support the inheritance of user and group data and provide a great deal of flexibility for defining the rights and roles for who can access which computers and what those users can do on the computers to which they have access.

How Inheritance Provides Additional Benefits

As discussed in Why use zones?, the primary benefits to using zones are:

- Identity management through user and group profile definitions
- Access and authorization control through rights and role definitions
- Delegated computer management for zone-based administrative tasks

Hierarchical zones provide additional flexibility for each of these benefits. For example, because hierarchical zones allow inheritance, hierarchical zones enable you define partial profiles and use variables that can be substituted at run-time when a user accesses a specific computer in a particular zone. Hierarchical zones also enable you to define access control rules and delegate administrative tasks at any point in the zone hierarchy.

This flexibility makes planning for hierarchical zones a key component of a successful deployment.

How Many Levels Should You Use in the Zone Hierarchy?

There are no predefined limits to the number of zones that can be used in a zone hierarchy or the number of levels deep zones can be nested in the hierarchy you define. For practical purposes, however, it is recommended using a hierarchy similar to the following:

- One or more top-level **parent** zones that include basic profile information for all users and groups that access the UNIX, Linux, and Mac OS X computers.
- One to three levels of intermediate **child** zones based on natural access control or administrative boundaries.

At each level in the hierarchy, profile information and access controls are inherited from the zone above and either applied or overridden by the child zone settings. At the lowest level of the hierarchy, you can override profile attributes or role assignments on any individual computers using **machine override** settings, if needed.

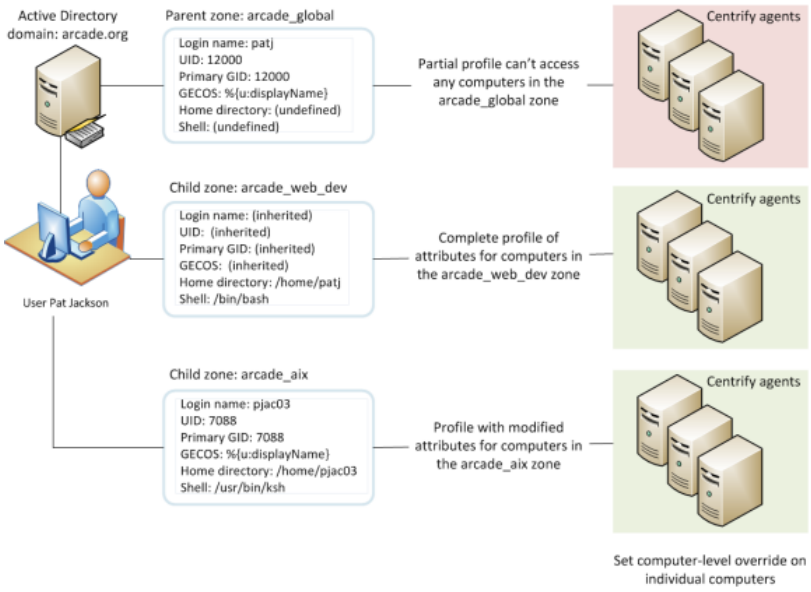
In addition, hierarchical zones support computer-based access rules, called **computer roles**, that enable you to selectively map a set of users with a particular role assignment access to a particular set of computers.

Identity Management and Inherited Profile Information

User and group profiles specify attributes such as the UID, primary group, home directory, and shell that are required for logging on to UNIX computers. You can specify all or part of the profile anywhere in the zone hierarchy, but users must have a complete profile to access computers they have permission to access. If the user or group profile is incomplete, it is invalid and ignored.

Working with Partial Profiles in the Zone Hierarchy

The profile information in the zone hierarchy is resolved from top to bottom for each user. For example, assume the user Pat Jackson has the login name patj and UID 12000 defined in the parent zone arcade_global and those profile settings are inherited without change, along with a default shell, home directory and other properties that are defined in the child zone arcade_web_dev. In a second child zone, arcade_aix, the UID for patj is set to 7088 to override the inherited UID. Changes to the profile properties can be made in any zone and inherited down the tree down to overrides set for specific individual computers, if needed.



Working with Variables in the Zone Hierarchy

Partial profiles enable you define a subset of profile attributes for users and groups that can be completed by lower level zones in the zone hierarchy. You can also define variables for resolving profile attributes. The variables are then substituted at run-time by adclient. For example, adclient can resolve the variable %{home}/%{user} to a platform-specific home directory for each user without having the attribute manually defined. You can set the variables at any level in the zone hierarchy, and they are inherited and resolved, or can be overridden, at a lower level in the tree.

You should note that variables can only be used to define profile attributes in hierarchical zones. You cannot import them or use them in classic zones.

Complete Profiles Do Not Grant Access

Creating user profiles in a zone does not give users access to any computers in the zone. The zone hierarchy simply creates a set of profiles with the potential to be granted access to computers. In previous versions of Server Suite software, enabling a UNIX profile for a user in a zone granted that user access to the computers in that zone by default. With hierarchical zones, the profile information only establishes the required properties for the user's identity, but does not grant access to any computers in any zones.

Access to computers is controlled through the definition of rights and roles.

Access Controls and the Assignment of Rights and Roles

A user must have a complete UNIX profile to log on to any computer in a zone. However, a complete profile alone does not allow a user to access any computers. The user must also have at least one role assignment that grants access somewhere in the zone hierarchy before any type of access is granted. Role assignments can be made anywhere in the zone hierarchy and inherited at a lower level in the tree.

Understanding Roles and Rights

Rights represent specific operations users are allowed to perform. A **role** is a collection of rights that can be defined in a parent or child zone and inherited. For example, a role defined in a parent zone can be used in a child zone, in a computer role, or at the computer level.

There are only a few predefined rights, called system rights. The system rights for Linux, UNIX, and Mac OS X are:

- **Password login and non password (SSO) login are allowed**: Specifies that a user is allowed to log on interactively using a password or without a password using a single sign-on token.
- **Non password (SSO) login is allowed**: Specifies that a user is allowed to log on using a single sign-on token.
- **Account disabled in AD can be used by sudo, cron, etc.**: Specifies that an account that is disabled is allowed to access the computer. This right enables service accounts that run without a password to perform operations.
- **Login with non-Restricted Shell**: Controls whether a user gets a full shell or is forced into a restricted shell. Users must be assigned at least one role with this right to have access to a standard shell environment. A restricted shell only allows a user to execute explicitly defined commands.

The system rights for Windows computers are:

- **Console login is allowed**: Specifies that users are allowed to log on locally using their Active Directory account credentials.
- **Remote login is allowed**: Specifies that users are allowed to log on remotely using their Active Directory account credentials.

In addition to the platform-specific system rights, there is a common system right that allows users to bypass auditing or role restrictions to log on when there are problems on a computer. The **Rescue rights** option allows you to specify the users who can log on if problems with the authorization cache or the auditing service on a computer are preventing all other users from logging on.

You grant users permission to access computers by assigning them to a role that includes one or more access rights. By default, zones only contain the following predefined roles to grant basic access rights:

- **UNIX Login** role allows users assigned this role to log on and access UNIX computers in the zone.
- **Windows Login** role allows users assigned this role to log on and access Windows computers in the zone.

There are additional predefined roles that grant specific rights, such as the right to log on if auditing is required but not available. The predefined roles exist in each zone, but their role names are qualified by the zone name so that the same role name in a parent zone and a child zone are considered different roles. For deployment, the predefined roles enable you to migrate existing users without developing custom role definitions. After deployment, you can define additional rights, roles, and role assignments to refine how users and groups access computers in different zones.

Working with a Candidate Set of Profiles

Ultimately, the purpose of the zone structure is to determine who has access, and what kind of access, to a computer. The candidate set of profiles that have the potential to access a computer is resolved by traversing the zone hierarchy from top to bottom. Because profile data is defined separately from the role assignments that control access, you can define an inclusive set of user profiles in a parent zone to create a candidate set that can then be applied to multiple child zones. In each child zone or at the individual computer level, you can use role assignment to control access for specific users from the inclusive candidate set.

Delegation in Hierarchical Zones

Hierarchical zones enable you to create a separation of duties for zone administration without recreating user and group profiles in multiple child zones. You can create full or partial profiles in the parent zone and inherit them into the child zone. Within each child zone, zone administrators can modify the profiles, as needed, and assign roles to control access to the computers they manage.

Designing a Zone Structure for Your Environment

Because the flexibility of hierarchical zones is a key element in designing the zone structure for your deployment, the next sections describe how to set up and use parent and child zones through sample deployment scenarios. The scenarios illustrate a basic deployment model, which will then be used to discuss how to migrate existing users and groups to Active Directory.

Your own zone structure and deployment model can be more complex than the one described in this guide. However, the deployment model described in the next sections is intended to ensure that you have a successful initial deployment. Over time, it is likely that you will change and adapt the zone structure to requirements that are specific to your organization. There are also multiple ways to accomplish the tasks described in the next sections of this guide. You can use other strategies and techniques for deployment if appropriate for your organization.

Preparing To Migrate Existing Users And Groups

This section describes how to prepare your environment for migration and computer access, including how to create and configure the initial zones. This chapter also describes how to import existing account information into Active Directory, set up provisioning for new users and groups, and configure role-based access controls.

Collecting And Analyzing Users and Groups

Before you create any zones, you should collect and analyze information about existing users and groups in the target set of computers. After you have investigated user and group attributes and identified invalid accounts and conflicting attributes, you can draft a basic zone design that addresses the needs of that user population. The goal of the initial zone design is to provide access to those users who currently have access, so they can transition to Active Directory with no disruption to their work. Later, you can refine access to computers through role assignments, filtering, and other options, if needed.

Collecting Information from Other Departments in Your Organization

Before you look at the content of identity stores you want to migrate, you should consider other sources of information that will help you identify a definitive set of legitimate users. For example, it can be useful to contact people in other departments who have reliable knowledge about the current organization or historical knowledge

about how the organization has evolved. Individuals with information about a segment of the user population can help you identify accounts that are obsolete or were created for testing, or belong to users who have left the company or moved to another department.

As a starting point for collecting information about existing users and groups, consider doing the following:

- Contact HR to get an up-to-date list of current active-duty employees, contractors, and consultants. You can use this information to compare personnel records to the UNIX accounts to be migrated. After you identify which accounts correspond to people in the organization, you can create a spreadsheet to record the UNIX user names, UIDs, and other useful fields for those accounts.
- Contact enterprise security administrators or department-level UNIX administrators to determine whether all of the accounts defined for a computer still need access to that computer. For example, you should determine if any users validated as current employees have changed departments or job functions. If a user no longer needs access to some computers, you may not need to add a profile for that user.
- Identify any conventions used in defining the namespace. For example, is there a standard format for the contents of the GECOS field? How do the conventions used for UNIX, Linux, or Mac OS X accounts compare to the conventions used in Active Directory? For example, is the convention used for the UNIX login name the same as the convention used for the user's sAMAcountName in Active Directory? Does the GECOS field follow the same conventions as the user's displayName in Active Directory?
- Identify which user attribute fields that can be used as primary keys for identifying a unique user. Depending on the conventions you use for creating new accounts, the user name, user identifier (UID), or the GECOS field may be a reliable field for identifying real users and mapping them to Active Directory accounts. If you use a standard provisioning convention across platforms for an attribute such as the GECOS field or user name, the convention makes it much easier to identify unique users and map user profiles to Active Directory accounts.

Using a Script to Retrieve User and Group Profiles for Each Computer

Alternatively, you can write a script to retrieve all of the /etc/passwd and /etc/group files in the target set of computers. For example, to create a hostname.passwd and hostname.group file for each computer in a target set of computers, you might use code similar to the following:

```
for name in cat hostname.txt; \
 do scp $name:/etc/passwd $name.passwd; \
 scp $name:/etc/group $name.group; \
 done
```

This sample script includes the computer host name in the file name, so that you can determine which user and group definitions came from which computer. If you use a script to collect user and group information, copy all of the files generated by the script to a common location for analysis.

Collecting Data from NIS Domains

If you're migrating users and groups from a NIS domain, you can use ypcat or niscat to generate a copy of the NIS passwd and group maps once for each NIS domain. For example, run commands similar to the following:

```
ypcat passwd > domainname.passwd
 ypcat group > domainname.group
```

If you are collecting user and group information from NIS maps, copy all of the files generated by these commands to a common location for analysis.

Identifying Accounts that Should Not Be Migrated

After you have collected information about the existing users and groups in the target set of computers, examine each of the passwd and group files and NIS domain maps to create a list of users and groups that you do not plan to migrate into Active Directory. For example, in most cases, there's no compelling business reason to migrate default system accounts, such as nfsnobdy or games, that will not map to Active Directory users. You should also eliminate accounts for people who have left your organization, and accounts that are locked or obviously invalid.

Eliminate Default System Accounts

In most cases, you can ignore all UNIX users with a UID less than 99 because those are the default operating system accounts. You may also want to skip migration for some or all UNIX service accounts unless you explicitly want to manage those service accounts, and any privileged commands they run, through Active Directory and zones.

You can manage the passwords for UNIX service accounts using Access Manager without having those accounts defined in zones or in Active Directory. Therefore, you may want to leave most or all of the service accounts as locally defined accounts.

In general, the only reasons to migrate default system or service accounts to Active Directory are:

- If you want to use Active Directory password policies for the account.
- If the service account itself owns one or more privileged commands that you want to manage through Centrify role definitions rather than locally in the sudoers file.

Typically, only service accounts that own special permissions, such the oracle user account, are migrated to Active Directory.

Remove Other Invalid Accounts

You should also skip migration for users who have left the organization and profiles that contain invalid data. Scan the data set for user accounts and groups that are locked or indicate they were created for testing. You should also check for profiles that contain obvious errors or legacy information no longer used.

In some cases, you may need to contact workstation or application owners directly to determine whether a profile should be skipped for migration. For example, assume the /etc/group file contains an entry for clowns with krusty, bozo, and tadams as members and there are valid user profiles for those three users. You may suspect the clowns group was created for some local testing, and therefore, not a candidate for migration. However, there's no definitive indication that the clowns group should be skipped without more information.

Create a List of the Users and Groups to Ignore

Add all of the accounts that should not be migrated to a text file. For example, create a text file named user.ignore and include all of the user accounts you don't want to migrate into Active Directory. For default system and service accounts that have known UIDs, you can create the text file programmatically using code similar to the following:

```
cat *.passwd ZZ_BAR_ZZ \egrep "x?:[0-9]:[0-9].ZZ_BAR_ZZx?:[0-9][0-9]:[0-9].ZZ_BAR_ZZx?:60[0-9][0-9][0-9]:[0-9]ZZ_BAR_ZZx?:65[0-9][0-9][0-9]:[0-9]" ZZ_BAR_ZZ \
 cut -d ":" -f 1 | \
 sort | \
 uniq | \
 sed 's/^/\^/' > user.ignore
```

Other accounts you have identified as invalid can be added manually or using code if they share some common attribute, such as LOCKED in the password field.

Analyze User Profiles for Conflicting Attributes

After the initial analysis to remove profiles that should not be migrated, you have a candidate data set of users and groups to import. The next step is to analyze the attributes in user profiles to identify any potential problems that you will need to address when you move the profiles into zones. Delinea Professional Services offers scripts that assist in this process. If you are analyzing the files manually or writing your own scripts, here are the common issues you need to check for:

- Determine whether any user name has more than one UID in the target set of computers. The UID is the primary way of determining file ownership and file permissions. On a single UNIX system, a user can only have one UID. However, across all of the computers in the target set, the same user name might have more than one UID.
- Determine whether any UIDs are associated with more than one user name.
- Determine whether any users have other profile conflicts, such as more than one primary GID, home directory, or shell on the computers in the target set.

In doing your preliminary analysis, keep in mind that you need to know which user profiles are associated with which people in your organization. For example, do the user names ldavis and davle refer to the same person (Len Davis) or to different people (Len Davis and Leslie Davidson).

This analysis of existing user profiles will help you identify the requirements of your initial zone design. The zone design allows you to use conflicting attributes as-is, without modifying any of the legacy data. You need to be aware of where there are conflicts so you can address them, but you do not need to change values for any attributes.

Analyze Group Profiles for Conflicting Attributes

You need to perform a similar analysis across the groups in the target set of computers. Delinea Professional Services offers scripts that assist in this process. If you are analyzing the files manually or writing your own scripts, here are the common issues you need to check for:

- Determine whether any group name has more than one GID in the target set of computers. Like the UID, the GID affects file ownership and file permissions. On a single UNIX computer, a group name can only have one GID. However, across all of the computers in the target set, the same group name might have more than one GID.
- Determine whether any GIDs are associated with more than one group name.
- Determine whether any group has a different set of members on any computers in the target set. Group membership is particularly important for zone design. The members of a group must be consistent across all of the computers in a zone.

As with the user analysis, this analysis of existing group profiles will help you identify the requirements of your initial zone design. The zone design allows you to use conflicting attributes as-is, without modifying any of the legacy data. You need to be aware of where there are conflicts so you can address them, but you do not need to change values for any attributes.

Create a Working Set of User and Group Profiles

After you have identified profiles that should not be migrated and noted any conflicting attributes you need to address, you have a known set of user and group profiles that you plan to migrate into Active Directory. The next step is to remove the users who should be ignored from list of users to import, and to remove the groups that should be ignored from the list of groups to import. You can do this manually or write a script that removes the profiles defined in the user.ignore and group.ignore files and outputs the results to a new file. For example, you might use code similar to the following to remove ignored users and generate a working set of user profiles:

```
mkdir output; \
 for name in cat hostname.txt; \
 do egrep -v -f user.ignore $name.passwd > \output/$name.passwd; \
 done
```

How Migration Affects the Zone Design

As discussed in Why use zones?, identity management is one of the primary benefits of using zones. Identity management is important because most organizations have an existing user population where users can have multiple UIDs or other attributes, such as different default shells, on different computers and groups with the same name can have different members. Each user has one Active Directory user object but may have multiple UNIX profiles, some with attributes in common and some with different settings. Zones allow you to migrate the profile information as it is defined, setting overrides where necessary, so that you can manage and report on the accounts without rationalizing the user namespace.

For all of the computers in a zone, a user or group has one profile definition, but the user or group could have different profile attributes on the computers in a different zone. Hierarchical zones make the zone design even more flexible. Hierarchical zones allow you to define one or more profile attributes in a parent zone and use those profile attributes in all child zones. In practice, this enables you to define a dominant set of attributes in a parent zone, and inherit the common attributes in one or more child zones. You can also override any attribute at any point in the zone hierarchy down to an individual computer.

For example, if a UNIX administrator has a consistent profile across all of the UNIX computers, but a customized home directory on two Mac OS X computers, you could define the default profile for the user in a parent zone, then create a child zone for the Mac OS X computers and inherit all of the profile attributes except the home directory setting.

In planning the migration, you identify the attributes that are the same across the target set of computers and where there are differences. If you use hierarchical zones, the primary task is identify one or more potential parent zones. For example, if you are migrating two NIS domains, you might create two parent zones because the UID space is unique in each domain but there would be conflicting attributes if the domains were combined into a single parent zone. The computers in each of the parent zones would inherit the UID and other profile attributes from each distinct NIS domain.

After you have identified one or more parent zones, you can plan how you will use child zones and overrides to manage profile attributes, implement access controls, and delegate administrative duties.

Creating the First Zone

It is recommended that you plan to use hierarchical zones and create at least one top-level parent zone. A single top-level zone for your organization is also useful for long-term management of UNIX profiles. You can have more

than one top-level parent zone. For example, if your organization has subsidiaries that are run independently or distinct geographical locations managed by different teams, you may want to create separate parent zones for those lines of business or locations.

Having a single top-level parent zone enables you to create an administrative group of super-users who can log on to every UNIX computer in your organization. It also allows to define some common rights and roles that can be inherited by child zones and the computers in those zones. Having a global or master zone for the entire organization also simplifies setting up provisioning for new accounts. However, there's no restriction on the number of parent or child zones you create. If you have a distributed environment and delegate administrative authority to separate teams, you can create as many parent zones as you find useful.

This guide describes how to set up the migration environment using one top-level parent zone. If you create more than one parent zone, you may need to repeat steps or extrapolate additional steps from the information presented here.

Create a Top-level Parent Zone

Before you migrate users and groups or add computers to the domain, you must have at least one zone. It is recommended that you create one top-level parent for your organization, which is similar to having a single forest root domain.

To create the top-level parent zone

1. Log on to the Windows computer where Authentication & Privilege is installed and open Access Manager.

   If you are not currently connected to the appropriate forest, specify the domain controller to which you want to connect.

2. In the console tree, select **Zones** and right-click, then click **Create New Zone**.
3. Type the zone name and, optionally, a longer description of the zone.

   In most cases, you should use the default parent container and container type that you created when you configured the Active Directory forest, and the default zone type, which creates the new zone as a hierarchical zone, then click **Next**.

   The only reasons for changing the default settings would be if you want to:

   - Create a zone in a new location to separate administrative activity for different groups of administrators.
   - Create zones as organizational units because you want to assign group policy objects to zones.
   - Create a classic zone for backwards compatibility or are using the Microsoft Services for UNIX (SFU) schema.

     For additional details about any of the zone fields, press F1 to view context-sensitive help.

4. Review the information about the zone you are creating, then click **Finish**.

Add Provisioning Groups to the Parent Zone

The next step in configuring the top-level parent zone is to create two Active Directory groups that will enable automated provisioning and de-provisioning of users and groups in the toplevel parent zone. By creating these provisioning groups in the parent zone, you can integrate the provisioning of UNIX users and groups with your existing processes for provisioning Active Directory users.

The provisioning groups are not required for migration, but a recommended configuration for the top-level parent zone you are creating as the first zone in the environment.

It is recommended you follow the naming conventions suggested for these groups. If you use a different naming convention, you should be sure it is well documented in your internal process documentation.

To add the provisioning groups for user and group profiles to the parent zone

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Select Provisioning Groups, right-click, then select **New > Group**.
4. Type the group name using the format ZoneName_Zone_Groups. For example, if the zone name is arcadeGlobal, type arcadeGlobal_Zone_Groups, then click **OK**.

   The Zone Provisioning Agent will use this group when processing the business rules for adding or removing group profiles in the parent zone.
5. Select Provisioning Groups, right-click, then select **New > Group**.
6. Type the group name using the format ZoneName_Zone_Users. For example, if the zone name is arcadeGlobal, type arcadeGlobal_Zone_Users, then click **OK**.

   The Zone Provisioning Agent will use this group when processing the business rules for adding or removing user profiles in the parent zone.

To prevent problems in UIDs and GIDs for existing users and groups, you should import existing user and group profiles before defining the business rules for automated provisioning of new accounts. After you complete the migration of the existing user population, you will define the business rules for the ZoneName_Zone_Groups and ZoneName_Zone_Users groups you just created.

Create Groups for the Default Roles in fhe Parent Zone

The next step in configuring the top-level parent zone is to create two Active Directory groups for the default listed and UNIX Login roles that are predefined in hierarchical zones.

- If you have a single top-level parent zone, users with a listed role can be recognized as having a valid profile on every UNIX computer in the organization. However, users in the listed role are not allowed to log on to any of those computers.
- If you have a single top-level parent zone, users with a UNIX Login role can log on to every UNIX computer in the organization.

For the toplevel parent zone, the UNIX Login role is intended for enterprise-level systems administrators who need to be able to log on to any UNIX computer in the organization. Because these are powerful roles in the parent zone, only a limited number of users would ever be assigned to these roles. However, the listed and UNIX Login roles are key components of migration when you create one or more child zones. If no users in the organization will be assigned these roles in the parent zone, you can skip the creation of the Active Directory groups for roles in the parent zone.

To create the groups for listed and UNIX Login roles in the parent zone

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Select the User Roles organizational unit, right-click, then select **New > Group**.
4. Type the group name using the format ZoneName_Role_RoleName. For example, if the zone name is arcadeGlobal, type arcadeGlobal_Role_Listed, then click **OK**.
5. Select the User Roles organizational unit, right-click, then select **New > Group**.
6. Type the group name using the format ZoneName_Role_RoleName. For example, if the zone name is arcadeGlobal, type arcadeGlobal_Role_Login, then click **OK**.

Delegate Administrative Tasks on the Parent Zone

The next step in configuring the top-level parent zone is to delegate administrative authority to the Zone Administrators group and to delegate specific permissions to the service account for the Zone Provisioning Agent to enable automated provisioning of user and group profiles in the parent zone.

To delegate administrative tasks on the top-level parent zone

1. Start Access Manager.
2. In the console tree, expand the **Zones** node.
3. Select the top-level parent zone, right-click, then click **Delegate Zone Control**.
4. Click **Add**.
5. Change the Find list from User to **Group**, type z, then click **Find Now**.
6. Select Zone Administrators in the results, then click **OK**.
7. Click **Next**.
8. Select **All** to enable members of the Zone Administrators group to perform all administrative tasks on the top-level parent zone, then click **Next**.
9. If you are delegating the task of joining computers to a zone, you can specify the scope of computers you can join to the zone; you pick a container in Active Directory to grant access to.

   If you leave the scope blank, the scope is the domain root. Be aware that the postalAddress field is used for information about joining computers to a zone; if you lookup the permissions for people you've delegated the task of joining computers to a zone, they'll have permissions to the postalAddress field for the affected computers.
10. Review your selections, then click **Finish**.
11. Right-click, then click **Delegate Zone Control**.
12. Click **Add**.
13. Type all or part of the service account name for the Zone Provisioning Agent that you created in About Zone Provisioning Agent and its requirements, click **Find Now**, then select the service account in the results and click

OK.

14. Click **Next**.
15. Select the following delegation rights for the Zone Provisioning Agent service account, then click **Next**:
    - Change zone properties
    - Add users
    - Add groups
    - Remove users
    - Remove groups
16. Review your selections, then click **Finish** to save the changes and close the dialog.

Link a Role Group to a Role Assignment in the Parent Zone

The next step in configuring the top-level parent zone is to link the Active Directory role groups created in Create groups for the default roles in the parent zone with the listed and UNIX Login role definitions that are predefined in the parent zone. You create this link between an Active Directory group name and the combination of rights associated with a role name by assigning the Active Directory group to the role.

1. Start Access Manager.
2. In the console tree, expand **Zones**, the top-level parent zone, and Authorization nodes.
3. Select Role Assignments, right-click, then click **Assign Role**.
4. Find the ZoneName_Role_Listed Active Directory group, then click **OK**.
5. Click **Browse**.
6. Select the listed role from the list of available roles, then click **OK**.
7. Check that the Start immediately and Never expire options are selected and appropriate or deselect those options and set start and end times, then click **OK**.
8. Repeat Step 3 through Step 7for the ZoneName_Role_Login Active Directory group and the UNIX Login role.

Create One or More Child Zones

After you have created a parent zone and prepared it with provisioning and role groups, you can create one or more child zones. You can create the child zones based on any logical model you choose. This is where the analysis of common and conflicting attributes and some creativity come into play.

Logical Models for Defining Zones

Because the zone design uses hierarchical zones, you can override attributes at any zone or computer level to deal with conflicts in legacy profile data. With this flexibility, you can experiment with different possible designs, for example, based on delegated administrative authority or physical location. Some common models for grouping a set of computers, users, groups, roles, and rights in the same zone include:

- **By shared identity store** For example, existing identity stores, such as NIS domains or a centralized user and group database, often provide a natural boundary for zones. This strategy is especially effective if each identity

store has a consistent namespace, without profile conflicts. It is less effective if the computers that share a common administrative group use local /etc/passwd and /etc/group file to store account information.

- **By application or function** For example, you might want to groups all of your database servers or web farm servers into their own zones. As part of this design, you might need to evaluate whether you are combining development computers with production servers and what role assignments you'll need to control what users can do on each type of computer.
- **By geographical region or line of business** For example, all of the UNIX computers that support a business unit could be logically grouped together. As part of this design, you might evaluate whether different business units should be responsible for provisioning users or assigning roles within their own business unit.
- **By host name** If you already have a meaningful host name convention that identifies machine owners or primary function, you may want to create zones based on that naming convention.
- **By platform and operating system** You can use this strategy, for example, to create separate zones for Red Hat Linux workstations and Sun Solaris UNIX workstations.
- **By department or user community** You can use this strategy, for example, to create separate zones for the computers that host financial applications and computers used by software developers.

You are not required to create child zones. You could control access to the parent zone through role assignments. For most organizations, however, one or more child zones makes it easier to assign roles and manage group membership.

Depending on your target set of computers, you may decide to start with one or two child zones or skip the creation of a child zone.

Create a Child Zone under the Parent Zone

Creating a child zone is similar to creating a parent zone. You select the parent in the left pane, then create and configure the child zone to prepare an environment into which you will migrate existing users and groups.

To create a child zone under the parent zone

1. Start Access Manager.
2. In the console tree, expand the **Zones** node.
3. Select the top-level parent zone, right-click, then click **Create Child Zone**.
4. Type a name and description for the child zone, then click **Next**.

   For example, if you are organizing by functional group, this zone might be finance or engineering. If you are organizing by data center location, the child zone might be sanfrancisco or seattle.
5. Click **Finish** to complete the zone creation.

The new zone is listed under the Child Zones node in the left pane.

Create Role Groups for Child Zones

The next step in configuring the child zone is to create two Active Directory groups for the default listed and UNIX Login roles that apply to this zone.

- In the child zone, users with a listed role can be recognized as having a valid profile but only on computers that are joined to the child zone. Users in the listed role for the child zone cannot log on to any of the computers joined to the child zone.
- In the child zone, users with a UNIX Login role are allowed to log on to every UNIX computer joined to the child zone if they have a UNIX profile for the zone.

For the child zone, the UNIX Login role is intended zone-level administrators and users who were previously able to log on to the UNIX computers joined to the child zone. The listed and UNIX Login roles are key components of migration when you create one or more child zones.

To create the role groups for listed and UNIX Login roles in the parent zone

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Select User Roles, right-click, then select **New > Group**.
4. Type the group name using the format ChildZoneName_Role_RoleName. For example, if the child zone name is sanfrancisco, type sanfrancisco_Role_Listed, then click **OK**.
5. Select User Roles, right-click, then select **New > Group**.
6. Type the group name using the format ChildZoneName_Role_RoleName. For example, if the zone name is sanfrancisco, type sanfrancisco_Role_Login, then click **OK**.

Delegate Administrative Tasks on the Child Zone

The next step in configuring the child zone is to delegate administrative authority to the Zone Administrators group. The steps are the same for the child zone as the parent zone, except that you expand the Child Zones node and select the name of the child zone before selecting the Delegate Zone Control command. You should still assign the Zone Administrators group All permissions.

You also have the option of assigning the permissions to join or leave to the Join Operators Active Directory group. If you pre-create computer accounts and allow the computer to join itself to the Active Directory domain, you can skip this step.

If you don’t want to pre-create the computer account and allow the self-service join, you must give members of the Join Operators group the following administrative tasks:

- Join Computers to the Zone
- Remove Computers from the Zone
- Modify Computer Profiles

Link Role Groups to Role Assignments in the Child Zone

The next step in configuring the child zone is to link the Active Directory role groups created in Create role groups for child zones with the listed and UNIX Login role definitions that are predefined in the child zone. You create this link between an Active Directory group name and the combination of rights associated with a role name by assigning the Active Directory group to the role. The steps are the same for the child zone as the parent zone,

except that you expand the **Child Zones** node and select the name of the child zone before selecting the **Authorization** node.

When you search for the Active Directory group to assign, you will select the ChildZoneName_Role_Listed, for example sanfrancisco_Role_Listed, for the listed role, and ChildZoneName_Role_Login, for example sanfrancisco_Role_Login, for the UNIX Login role.

Users who are added to the ChildZoneName_Role_Login group will be able to log on to computers that are joined to the child zone or any of its own children, but will not be able to log on to computers in other child zones.

Create Computer Objects For The Target Set Of Computers

When you manage UNIX computers with Centrify software, you add computer objects to Active Directory for those computers. These computer objects can be created automatically when a computer joins the domain, or created in Active Directory before the computer joins the domain. In most cases, Centrify recommends that you create the computer account objects before joining, if possible.

For deployment and migration, creating the computer objects before joining provides the following key advantages:

- You can define computer-level overrides before computers are added to the zone. This allows you to resolve issues with divergent UNIX profiles without having to change file permissions at the file system level.
- You can check who will have access to which UNIX computers before those computers join the Active Directory domain.

Pre-creating the computer objects enables you to check that you have user profiles and role assignments correctly defined before you join the UNIX computers to zones. Verifying this information before the join operation helps to ensure a smooth migration without disrupting users' access to files or applications.

Prepare A Computer Object Before Joining

In most cases, you should pre-create the computer object for every UNIX computer in every zone. For individual computers, you can use the Prepare Computer wizard to guide you through the process. However, you will probably want to create a Windows or UNIX script for performing the operation repeatedly. For example, you can use adedit or the Windows API to create a script.

To prepare a computer account in Active Directory using Prepare Computer

1. Start Access Manager.
2. In the console tree, expand the **Child Zones** node, then expand the child zone for this computer to join.
3. Select the Computers node, right-click, then click **Prepare Computer**.
4. Accept the default preparation options, then click **Next**.
5. Accept the default to **Create a new computer object**, then click **Next**.
6. Type the name of the computer object to create and modify the DNS host name of the computer object, if necessary.

   The computer name is the name of the computer principal in Active Directory. The DNS name is how the UNIX computer is currently registered in DNS. If you have a disjointed DNS namespace, you should be sure the DNS name is the name used in the computer's DNS entry.

7. Click **Change** and navigate to the organizational unit for storing computer principals. For example, if you created the organizational unit structure described in Creating recommended organizational units, select UNIX Servers and Workstations and click **OK**, then click **Next**.
8. Select an option for joining the computer to the domain, then click **Next**.
   - If you want to require users to interactively join the computer to Active Directory, click **Browse** to select the Join Operators group.
   - If you want to allow the computer to join itself to the zone, select **Allow the computer to join itself to the zone**. This option automatically associates the computer with the correct zone, so there's less chance of a human error.
9. Click **Browse** to select the Zone Administrators group, then click **Next**.

   With this setting, users in the Zone Administrators can override any inherited attributes of a UNIX user or a UNIX group profile on the computer.
10. Review your selections, then click **Next** to create the computer account.
11. Click **Finish** to complete the process.

You have now finished preparing the environment for migration and are ready to begin importing groups and users and assigning them appropriate roles.

Migrating Existing Users To Hierarchical Zones

Now that you understand how zones are used and have prepared an environment for an initial migration, you are ready to import the existing users and groups that you have identified as candidates for being migrated to Active Directory.

This section uses a sample data set to illustrate how to migrate an existing user population into hierarchical zones and how to assign the appropriate roles to convert from a legacy authentication model to Active Directory and Server Suite.

Importing Group Profiles

After you have created one or more zones and separated the users and groups to ignore from the users and groups that you think should be migrated to Active Directory, you must decide which groups apply to which zones. For example, if you have some groups with the same group profile and group membership in all zones, you would import those groups into the top-level parent zone so that they also exist, with the same definition, in all child zones. If a group is only applicable for computers in a child zone, you can import the group profiles directly into that zone. You can also override group profile attributes on specific computers, if needed.

After you have made these decisions, importing the groups is a simple process using either the **Import from UNIX** wizard or ADedit scripts with two important considerations:

- Group names must be unique in Active Directory. If you create a group with a common name, such as admins, you cannot create another group with the same name.
- Having the same UNIX group name on computers in different zones can create group collisions and inadvertent privilege escalation or file ownership conflicts.

To prevent group name collisions, Centrify recommends that you include the zone name in the Active Directory group name. You may also want to add a suffix that identifies the group as an UNIX security group. In most cases,

you create the Active Directory group object for the UNIX group in the UNIX Groups organizational unit if you created the organizational unit structure described in Creating recommended organizational units.

You should import group profiles and create the corresponding Active Directory groups for those groups before you import users. If you import group profiles first, you can resolve secondary group membership for users immediately after you import user profiles.

Import Unix Groups that Apply to All Computers into the Parent Zone

If your organization has a default UNIX administrators group or security group that you want to be available on all UNIX computers, that group is a good candidate for importing into the parent zone. Other groups that might be candidates for the parent zone are special purpose UNIX groups that own sudoers permissions that apply to all UNIX computers or an auditing group that requires access to all computers.

If you have identified any common groups, use the Import from UNIX wizard or a script to import the UNIX groups that should be available for all computers into the top-level parent zone.

To Import Unix Groups Using the Import from Unix Wizard

1. Start Access Manager.
2. In the console tree, expand **Zones** and the top-level parent zone.
3. Select UNIX Data, right-click, then click **Import from UNIX**.
4. Click **UNIX configuration files**, then click **Browse** to locate and select the group file to import, then click **Next**.
5. Select the option to automatically shorten the UNIX name, if desired, then click **Next**.
6. Leave **Store in Active Directory** selected and click **Next**.
7. Select **Check data conflicts while importing**, then click **Finish**.

   This step places the profiles under Groups as Pending Import.
8. Select one or more group names that are Pending Import, right-click, then select **Create new AD groups**.
9. Click **Browse**, navigate to the UNIX Groups organizational unit and click **OK**, then click **Next**.
10. Click **Add a prefix to group name**, type the parent zone name and an underscore (*), select the group scope as Global, then click **Next**. For example, if the parent zone name is arcadeGlobal, use the prefix arcadeGlobal.*

    Optionally, click **Add a suffix to group name** and type a suffix that identifies the group as a UNIX security group, for example, _unix.
11. Review the information displayed, then click **Finish**.

For more information about importing groups, see the *Administrator's Guide for Linux and UNIX*.

Import Unix Groups that Apply Only to a Specific Zone into a Child Zone

From your initial analysis and zone design, you should also have a reasonable plan for groups that apply to specific child zones. Groups imported into a child zone are visible to all the UNIX computers in that zone, but not in other zones. For example, assume you have identified an application-specific group, ora_app01, that allows users to use a database, and the database application exists three computers. In your zone design, you decide those three computers should be a single child zone. In that case, you import the ora_app01 group profile into the child zone group because the database application group is only relevant to the UNIX computers in the child zone.

The steps for importing into a child zone are the same as for the parent zone, except that you select the UNIX Data under the child zone name in the console tree and specify the child zone name as the prefix for the Active Directory group name.

Import a Group Profile or Override Attributes on Specific Computers

In some cases, you may have a UNIX group that only exists on one computer in a zone or exists on more than one computer but has different attributes on different computers. You can use computer-level overrides to handle these cases. Computer-level overrides enable Zone Administrators to create and manage group profile attributes manually for individual computers.

To create a group profile for a specific computer

1. Use Active Directory Users and Computers to create an Active Directory group in the UNIX Groups organizational unit. If the group only applies to a specific computer, you may want to use the computer name as the prefix.
2. Start Access Manager.
3. Expand the console tree to display the individual computer object under the zone the computer will join.
4. Expand UNIX Data, select Groups and right-click, then click **Create UNIX Group**.
5. Click the attributes to define, type the appropriate values, then click **OK**.
   - Click **GID** to manually specify a GID for the group profile on the selected computer.
   - Click **UNIX group name** to manually specify a group name for the profile on the selected computer.

Avoiding Group Collisions When Using Computer-level Overrides

If you create group profile overrides on individual computers, you should make sure that the UNIX group name and GID are not being used by any other groups in the parent or child zone. If the group profile defined for the computer is the same as a group profile defined for a group in the parent or child zone, users who should only be able to access files on the local computer may be able to access files owned by the group defined for the parent or child zone. This can be a difficult problem to identify. For example, assume you have an Active Directory group named contract_admins, but you have used the UNIX group name admins and the same GID as a group in the parent zone. Any user who is a member of the contract_admins group in Active Directory is going to have the same GID as the parent zone's admins group. If that happens, members of the contract_admins group will have access to the same files as the admins group in the parent zone.

The only way to identify when this problem occurs is by running the following command for a user in the contract_ admins group:

id -a

Importing User Profiles

You can import user profiles into the parent zone or into child zones. If you import user profiles into the parent zone, all existing users will be included in the candidate set of users who have the potential to log on to all of the UNIX computers in the organization. However, they are not granted any access by default. Instead, the management of identity information, such as the user name, UID, and primary group, is separate from privilege management. Users

cannot access any UNIX computers until they are assigned a valid role with the specific permissions they need to be recognized, allowed to log on, or run specific commands.

Although you can import users into the parent zone without granting them access rights, you may prefer to import them into one or more child zones. By importing users into specific child zones, you can limit the scope of their potential access. In general, this option is applicable for the majority of your end-users and can apply to other users, such as database administrators, project managers, and contractors who won't ever need access to all the UNIX computers in the organization.

At this stage you should decide whether to give users the potential to access all computers in the organization, or only the computers in one or more specific child zones. After you import the user profiles, you will use the default listed and UNIX Login roles or custom roles to control access to the UNIX computers.

The steps for importing user profiles into a parent or child zone are essentially the same as importing groups. You can use the **Import from UNIX** wizard or ADedit scripts to import the profiles into one or more zones. The profile information for any user can be different in each zone. If the profile information is divergent on any computer within a zone, you can set computer-specific overrides for any or all attributes.

If you are importing users from a file, you can write a script that modifies the GECOS field to use the same format used in the Active Directory displayName attribute before importing so that users are automatically mapped to their corresponding Active Directory accounts. For example, if your convention for the GECOS field is first_name last_name (Jae Wilson) but the convention used in Active Directory is last_name, first_name (Wilson, Jae), you must manually map the UNIX user account to the Active Directory account. If you modify the format of the GECOS field before importing, the Import from UNIX wizard can automatically suggest a candidate for mapping the UNIX user to an Active Directory user, if an account exists.

After you import users, their profiles are placed under Users as Pending Import. If the user has an existing Active Directory account, you can select the user name, right-click, then select **Extend existing AD user**. If an Active Directory account does not exist, you can select the user name, right-click, then select **Create new AD users**. You can then use **Check Status** to resolve group membership for Pending Groups. This command adds the imported users to the appropriate Active Directory groups that have UNIX profiles in the zone to complete the first phase of the migration to Active Directory.

For more information about importing users and resolving group membership, see the *Administrator's Guide for Linux and UNIX*.

How Group Membership Works Within Zones

When a UNIX group profile is imported into a zone, its group name and GID are recognized by all computers joined to that zone. However, the group membership might vary by computer. For a user to be a member of the UNIX group, the user must:

- Be a member of the Active Directory group.
- Have a complete UNIX user profile defined somewhere in the zone hierarchy (in the parent zone, a child zone, or with computer-level overrides).
- Be assigned the listed role or the UNIX Login role somewhere in the zone hierarchy (in the parent zone, a child zone, or with computer-level overrides).

For example, assume the users Alison and Clyde are assigned the UNIX Login role for the Engineering zone. As discussed in Create role groups for child zones, that means they are also listed as members of the Engineering_Role_Login role group in Active Directory. Clyde is also a member of the denali project group in the Engineering zone and has a profile defined in the parent zone. Alison's profile is defined in the Engineering zone. If the denali project group (Engineering_Denali in Active Directory) is added to the Engineering zone, both Alison and Clyde can log on to computers in the Engineering zone, but only Clyde will be a member of the denali UNIX group in the Engineering zone.

Assigning Roles to Existing Users and Groups

You have now imported the existing user and group profiles for a target set of computers into Active Directory. This is one critical component of migration because users must have a valid UNIX profile, that is, a unique user name, UID, primary GID, home directory and shell, in a zone for them to be recognized as valid users. However, Server Suite separates UNIX profile management from UNIX privilege management. Users cannot log on to UNIX computers until they are assigned a role that allows them to log on to those computers.

As discussed in Access controls and the assignment of rights and roles, a role is a collection of rights and there are two default roles: the listed role and the UNIX Login role. As part of deploying Server Suite software with the least disruption to your environment, your existing users must be able to log on to the UNIX computers they currently use. That is the primary purpose of the UNIX Login role: to allow you to quickly give log on access to a set of users in one or more zones. The UNIX Login role in the parent zone is intended for enterprise administrators who need log on access to all computers. The UNIX Login role in the finance zone would be for those users who currently have interactive access to the limited number of computers in that zone and would expect to have that access after migration.

The listed role is intended for users who need a valid profile defined but do not need interactive log on access to the computers in a zone. For example, you assign the listed role to remote NFS users so that they have access to their files without the ability to log on and open a shell. You can also use the listed role to give users access to applications, such as ClearCase or Samba, that require a UNIX profile without the ability to log on locally or remotely. The listed role in the software-dev zone would be for those ClearCase users who need to be recognized on all computers in the zone so they can check files in and out.

The next step in the migration is to identify which users should be assigned to each role in each zone you have created.

Using Active Directory Groups for Roles

For most organizations, the most efficient way to manage role assignment is by adding users to Active Directory groups, then managing those groups. Therefore, for management purposes, a Centrify access role should always be linked to an Active Directory security group. The Active Directory groups that identify the users in specific Centrify user roles are stored in the User Roles organizational unit. All of the users in a specific role group will share a common set of rights under UNIX. You can then use machine-level overrides for handling edge cases for individual computers.

Adding Users to Role Groups

There are many different ways you can add UNIX user profiles to an Active Directory group. For example, you can manually select a UNIX profile in a zone, right-click, then select **Add to a group** or select groups under the Role Assignments node for the zone, and modify the group membership. In most organizations, however, you can leverage your existing provisioning process. If your current provisioning process involves managing a group in

Active Directory, whether it is through automated scripts or human processes, you can use the same process for provisioning UNIX users.

Migrating Existing Users Into The Unix Login Role In The Parent Zone

In Create groups for the default roles in the parent zone, you created Active Directory security groups for UNIX Login and listed roles in the parent zone. If you want to give all users the potential to log in to all UNIX systems, you can make them members of the parentZone_Role_Login group.

Users who are members of this group and have a complete UNIX profile in the parent zone can log on to all UNIX computers that are joined to the parent zone and all UNIX computers joined to the child zones of the parent zone. However, if you add users to the parentZone_Role_Login group in Active Directory, but do not define a UNIX user profile in the parent zone, those users will only be able to log on to the UNIX computers in the child zones where they have a UNIX user profile defined or the individual computers where you define machine-level overrides to give them a UNIX profile.

The default UNIX Login role associated with the parentZone_Role_Login group does not grant any additional privileges. It simply allows users to log on to UNIX computers. Therefore, one strategy for migrating users is to add them all to parent zone's Login role group. You can then control access based on where the user's UNIX profile is defined and control what the user can do using additional role assignments. For example, you may create custom roles to grant expanded UNIX privileges.

Migrating Existing Users into the Unix Login Role in Child Zones

If you define user profiles for most of your users in the parent zone, you should not make them members the parentZone_Role_Login group. Instead, you can add users to the appropriate childZone_Role_Login groups. All of your existing UNIX users who can currently log on interactively to existing UNIX systems should be added to one or more childZone_Role_Login groups. For example, users who currently have access to all of the computers in the Engineering zone should be added to the Engineering_Role_Login Active Directory group. If those users also have a UNIX profile in the parent zone or the Engineering zone, they will be able to log on to all of the computers in the Engineering zone. If a user only needs access to a specific computer in the zone, you can use a machine-level override to give the user access to that specific computer.

You can use the Access Manager console, Active Directory Users and Computers, ADEdit or custom scripts to add UNIX user profiles to the appropriate childZone_Role_Login groups. If possible, you should integrate this part of the migration with your existing provisioning process to ensure that future requests for UNIX role assignments use the processes that line of business personnel already understand.

Migrating Existing Users into the Listed Role in Child Zones

After you have assigned users who must be able to log on to the UNIX Login role, you should identify users who should be assigned the listed role to limit the number of users allowed to log on. The listed role is intended for existing UNIX users who have a UNIX user profile in one or more zones that you want to allow to be listed in getent output without the ability to log on to UNIX computers in those zones.

The listed role is most commonly used for users who access UNIX applications, such as ClearCase, or Samba, or an NFS-mounted file system, that require a UNIX profile. In practical terms, however, this role also allows you to migrate users you aren't sure have been authorized for access. With this role, the user profile is recognized but the user cannot log on locally or remotely.

You can use the Access Manager console, Active Directory Users and Computers, ADEdit or custom scripts to add the UNIX user profiles to the appropriate childZone_Role_Listed groups. If possible, you should integrate this part of the migration with your existing provisioning process to ensure that future requests for UNIX role assignments use the processes that line of business personnel already understand.

Keep in mind that the childZone_Role_Listed group affects all the UNIX computers joined to the specified child zone. Before you move a user to the childZone_Role_Listed group, you should check whether there are any computers in the zone that the user must be able to access to prevent accidentally locking the user out. You can use a machine-level override to grant the UNIX Login role on a specific computer, if needed.

Using a Computer-level Override for the Unix Login Role

You can also create computer-level overrides for the UNIX Login or listed role, if needed. This is not typically part of the migration process. However, if your initial analysis identified a zone where overrides would be useful, you can include overrides in your migration plan. For example, assume you have a zone where most of the user profiles are common across a set of computers. If you import the UNIX profiles for that user population, you see that two users would have access to a UNIX computer where they previously did not have access. To preserve the existing access while migrating from the legacy environment, you can define the UNIX profile in the zone but control access for those two users with a computer-level override.

Managing Role Assignment Without Role Groups

You are not required to use Active Directory security groups to manage role assignments. You can manually add users and groups to roles within any zone. Manually adding a user or group to a role without using Active Directory groups makes integration with provisioning systems more difficult, however. Most identity management and provisioning systems are designed to work with Active Directory groups inherently. Therefore, associating Active Directory groups with Server Suite roles typically provides easier integration with existing provisioning processes.

If you decide to manually manage role assignments, you can use the Server Suite Access Module for Windows PowerShell, Server Suite Access SDK, or ADedit to create scripts that manipulate the objects in Active Directory. Role assignments are stored in Active Directory using Microsoft Authorization Manager containers. If you want to add and remove user and group assignments, you will need to develop custom code to accomplish those tasks.

Verifying Effective Users On Each Zone

Now that you have imported profiles and assigned existing users to the appropriate roles, you can verify who has access to the computers in each zone before you proceed with joining a domain. Checking the Effective Users in each zone enables you to verify the users who have been assigned the UNIX Login and listed roles before any users are affected by the changes.

You should have a checklist of the users who require interactive access on the computers in the target set and which user profiles you suspect only need to be recognized without the ability to log on. You can then using the Effective Users option to see the role assignments for the pre-created computer objects in the target set of computers. By comparing the list of users to the role assignments, you should be confident that you are ready to complete the migration by joining UNIX computers to the Active Directory domain.

Performing this step before joining the domain helps to ensure the transition to Active Directory does not interfere with end-users daily work or the delivery of business services. Therefore, verifying UNIX Login and listed access before joining computers to the domain is a key part of a successful migration.

To access the Effective Users for a zone

1. Start Access Manager.
2. In the console tree, expand **Zones** and the top-level parent zone.
3. Select a zone, right-click, then click **Show Effective UNIX User Rights**.
4. Review the list of UNIX user profiles for the zone in the UNIX users section.
5. Select a user name to display additional information about each user:
   - **Zone Profile** displays details about inherited profile attributes. For existing users being migrated, the profile attributes are typically explicitly defined. If a profile is defined higher up in the zone hierarchy, the Inheritance tab indicates where the profile attributes are defined.
   - **Role Assignments** lists the role assignments for the selected user in the zone. For the initial migration, users must be assigned the UNIX Login or listed role.
   - **PAM Access** lists the specific PAM application access rights associated with the roles a user is assigned. For example, the default UNIX Login role has the login-all PAM access right, which enables PAM authentication for all computers in the zone.
   - **Commands** lists the specific UNIX command rights associated with the roles a user is assigned. For example, you can define a role that allows users to run specific privileged commands as root. You can click the Commands tab to see the specific privileged commands defined for the role.
   - **SSH Rights** lists the specific secure shell (ssh) command rights associated with the roles a user is assigned.
6. Click **Close** when you have finished checking role assignments for the users in target computer of computers.

You can also select **Show Effective UNIX User Rights** for individual UNIX computers and generate Hierarchical Zone reports that describe the effective rights for computers and users.

Adding Existing Users and Groups to Provisioning Groups

After you have added the existing users and groups to the appropriate Login and Listed role groups, the next step for completing the migration to Active Directory is to add the existing user and group profiles to the Provisioning Groups you created for the parent zone. This step is not directly related to data migration, but enables you to prepare the environment for automated user and group fulfillment using on the Zone Provisioning Agent.

Add Existing Users To The Provisioning Group For The Parent Zone

At this point, you have imported legacy data into one or more child zones and accepted divergent profile attributes using computer-level overrides. You should now add all of your imported UNIX users to the provisioning group in the top-level parent zone. Adding users as members of the provisioning group will enable the Zone Provisioning Agent to define a new “universal” UNIX profile for legacy users based on business rules you establish for the parent zone. The new profile will not affect the existing file ownership, but will make it easier to provision and deprovision users moving forward.

As discussed in Installing Zone Provisioning Agent, the Zone Provisioning Agent enables you to define business rules for creating new UNIX profiles for new UNIX users. After you complete the migration and enable the Zone Provisioning Agent, it runs at a regularly scheduled interval to determine whether there are new users or users who should be removed. At each interval, the Zone Provisioning Agent compares the members of the parent zone’s Users provisioning group with the user profiles currently defined for the zone.

If there are UNIX profiles for users who aren't members of the provisioning group, the Zone Provisioning Agent removes those user profiles. To prevent the Zone Provisioning Agent from removing the imported data, you must add the Active Directory users associated with the imported user profiles to the parent zone's Users provisioning group.

To add existing UNIX users to the provisioning group for the parent zone

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Expand the Provisioning Groups organizational unit, then select the parentZoneName_Zone_Users group. For example, if the parent zone is arcadeGlobal, select arcadeGlobal_Zone_Users, right-click, then select **Properties**.
4. Click the **Members** tab, then click **Add**.
5. Search for and select the imported user accounts that you have mapped to Active Directory users, then click **OK**.
6. Click **OK** to save the provisioning group and close the Properties.

Add Existing Groups to the Provisioning Group for the Parent Zone

As with imported users, you should also add all of your imported UNIX groups to the provisioning group in the top-level parent zone. Adding the group profiles as members of the top-level provisioning group will enable the Zone Provisioning Agent to define a new "universal" UNIX profile for each group based on business rules you establish for the parent zone. The new profile will not affect the existing file ownership, but will make it easier to provision and deprovision users moving forward. Adding the UNIX group profiles to the top-level parent zone ensures that the Zone Provisioning Agent does not remove the imported groups from the zone.

To add existing UNIX groups to the provisioning group for the parent zone

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Expand the Provisioning Groups organizational unit, then select the parentZoneName_Zone_Groups group. For example, if the parent zone is arcadeGlobal, select arcadeGlobal_Zone_Groups, right-click, then select **Properties**.
4. Click the **Members** tab, then click **Add**.
5. Search for and select the imported user accounts that you have mapped to Active Directory users, then click **OK**.
6. Click **OK** to save the provisioning group and close the **Properties**.

Joining Computers to a Domain and Zone

You have completed the preparation of the environment and added existing users and groups to Active Directory. The steps up to this point have not affected the day-to-day activities of any UNIX users or groups, and have not

changed the configuration of any UNIX computers. The final step in the migration requires you to join UNIX computers to the Active Directory domain. This step does have the potential to affect end-users.

This section describes how to complete the migration by joining the target set of computers to an Active Directory domain and a Server Suite zone.

Using Adjoin on New Computers

You can run the adjoin command interactively or in a script to join UNIX computers to Active Directory. One advantage to using the adjoin command is that it enables you to add the join operation to the steps for building a new UNIX computer. For example, if you have a process for provisioning a new UNIX computer, you can add an adjoin step that allows the new UNIX computer to join itself to Active Directory. Provisioning new computers to join the domain when they are built ensures that there are no new local users being defined on those UNIX computers.

Running Adjoin Requires Unix and Active Directory Privileges

On UNIX, running `adjoin` requires you to log on as root, be a member of the wheel group, or have root equivalent privileges in the `sudoers` file. On Mac OS X computers, `adjoin` requires the administrator account and password.

Specifying the Required Options

The basic syntax for the adjoin command is:

```
adjoin [options] domain_name [--zone zone_name | --workstation]
```

The `domain_name` should be a fully-qualified domain name; for example, sales.acme.com. If you are using adjoin to provision new computers, there are several options you should specify on the command line or in the script.

- Use the `--container` or `-c` option to specify the location for the computer account. Typically, you should use the organizational unit that you created for UNIX Servers and Workstation under the top-level UNIX organizational unit. It must be the location you used when you pre-created the computer object. For example:

  `-c "ou=UNIX Server and Workstations,ou=UNIX"`
- Use the `--selfserve` or `-S` option to specify that you want the computer to join itself to the Active Directory domain.
- Use the `--zone` or *-z* option to specify the name of the zone to join. You must specify a zone name unless you are joining Auto Zone using the `--workstation` option.
- If you have a disjointed DNS environment where the Active Directory domain for the computer account does not match the name of the DNS domain, you must also specify the -name and `--alias` options. The `--name` option specifies the name of the Active Directory computer object and the `--alias` will be the fully-qualified DNS name of the computer.
- Use the `--computerpassword` or `-X` to specify the password of the precreated computer account. You must also specify either `--precreate` or `--selfserve`. If you don't specify the password, the default password will be used.

For example, update your provisioning process for a new computer to include a command similar to the following:

```
adjoin -c "ou=UNIX Server and Workstations,ou=UNIX" -S -z production arcade.net
```

For complete information about adjoin options, see the adjoin man page.

Pre-staging Before Using Adjoin on a New Machine

When joining a large AD environment, the join procedure can take a very long time -- up to dozens of minutes. This becomes a concern in some use cases, such as starting an Amazon EC2 instance that needs to join the domain to provide service.

To speed up the adjoin process, the adjoin `--prestage` option uses existing cache files instead of populating cache from scratch.

Some preparation is required to take advantage of the `--prestage` option:

- Prepare a pre-staged cache directory on a joined machine
- Copy the cache directory to the new machine

Security Requirements

To use the `--prestage` option, ensure the following:

- Joined and new machine requirements:
  - The `--prestage` option can only be used between machines that have the same platform, architecture, and Authentication Service (Centrify DirectControl) release version installed.
  - Adclient cache data encryption feature cannot be enabled on the joined machine. See the adclient.cache.encrypt parameter.
- Pre-staged cache directory on joined machine requirements:
  - On a joined machine, create or designate a directory for the pre-staging cache files.
    - The directory must be in a safe path. That means all levels of parent directories are owned by system accounts.
    - The directory cannot be either group or world writable.
- Content for the pre-staged cache directory on the joined machine:
  - Place the cache files (dz.cache, dc.cache, gc.cache,.idx and kset. files) in the specified directory.
  - Ensure the cache files are owned by system accounts.
  - Files cannot be either group or world writable.
  - Symlink is not allowed for the cache files.
- Zone hierarchy changes are not allowed between the staging directory and the new machine. This includes:
  - zone name change
  - zone GUID change
  - zone schema change

Preparing to Use the --prestage Option

1. Create a directory on a joined machine. For example, /pre.
2. Stop `adclient` on that machine.

3. Copy the `/var/centrifydc/` directory to the pre-staged directory on the joined machine.

   For example:

   Copying the `/var/centrifydc/` directory to the pre-staged directory, /pre, places a copy of the required files in `/pre/centrifydc/`.

4. Verify the pre-staged directory on the joined machine contains all the .idx, .cache, and kset. files.
5. Copy the pre-staged directory to the new machine.

   Use a method of your choice, such as scp or sftp.

   This is done so the pre-staged files are available locally on the new machine.

6. Add the option to the adjoin command when adding the new machine. The syntax is:

   ```
   -E | --prestage <directory>
   ```

   where directory is the path to the pre-staged directory on the new machine.

   For example, if the pre-staged files are in directory, `/pre/centrifydc/`, use the following adjoin command.

   ```
   adjoin -z <zone> -E /pre/centrifydc<domain>
   ```

Verify Authentication After Joining the Domain By Logging On

As the final step in the initial migration, you should verify that authentication for an Active Directory user is successful. You can do this by logging on to the UNIX console using either the UNIX user name or the Active Directory User Principal Name for a user assigned to the UNIX Login role. When prompted, type the Active Directory password for the account. If you are able to log on using the Active Directory password, you know that authentication is being handled by Active Directory and the user account has been successfully migrated.

You should also verify that you can log on remotely using a secure shell (ssh) connection and that you can use other services such as ftp.

If users have trouble logging on after a UNIX computer has joined the domain, it is typically because they're not assigned the UNIX Login role or don't have a valid UNIX profile in the zone. You can use the Show Effective UNIX User Rights command to check which users have profiles and what roles have been assigned to users who have access to the selected computer.

Provisioning New User and Group Profiles After Migration

After you have completed the basic migration for a set of existing users and groups, you can continue with the Centrify deployment by configuring the environment for automated provisioning of new users and groups. At this stage, you have already built the foundation for the automated addition and removal of users and groups. The next steps involve defining the business rules for creating new user and group profiles. The goal of this section is to help you identify and integrate a provisioning process for new UNIX users and groups.

Integrating with Existing Provisioning Processes

The Zone Provisioning Agent and the provisioning groups you created in Add provisioning groups to the parent zone are intended to integrate the provisioning of UNIX users and groups with your existing account fulfillment process. Those groups enable you to leverage existing processes because most organizations have well-defined and standardized procedures for provisioning new Active Directory users based on Active Directory group membership.

If possible, you would like to use the same or a similar process for provisioning UNIX users and groups. If you can integrate the provisioning of UNIX users and groups with your existing process, the people in your organization can use tools they are familiar with and won't have to learn an entirely new process.

However, defining the business rules for adding new user and group profiles to zones requires some planning. In particular, you need to make decisions about Active Directory group membership, primary group definitions for users in zones, and how profile attributes are defined.

Defining the Business Rules for New Groups in the Parent Zone

You have already started the process of integrating the provisioning for UNIX users and groups when you added imported accounts to the Active Directory provisioning groups in Adding existing users and groups to Provisioning Groups. The next step is to define the business rules for creating new UNIX group profiles in the top-level parent zone.

> The business rules you define only affect new UNIX user and group profiles. The imported legacy data remains unchanged, and the Zone Provisioning Agent will not modify any attributes on the existing user and group profiles.

Configure the Business Rules for Automated Provisioning of Group Profiles

You configure the business rules for automated provisioning of group profiles on a zone-by-zone basis. When you use hierarchical zones, you typically want to configure the business rules for the parent zone so that the profile can be inherited in all child zones. Remember that the profile, by itself, does not provide any access to the computers in the child zones, and that you can override any inherited attributes in any zone or on individual computers.

To Configure the Business Rules For Groups in the Parent Zone

1. Start Access Manager.
2. In the console tree, expand the **Zones** node.
3. Select the top-level parent zone, right-click, then click **Properties**.
4. Click the **Provisioning** tab.

   If you are defining business rules for a parent hierarchical zone and want to establish a “source zone” for profile attributes, click **Advanced**. You can then select the Source zone for any or all user and group profile attributes. If you select Source zone for any attribute on the Advanced Provisioning page, you can click **Browse** to search for and select the zone to use as the source zone. In most cases, selecting a source zone is not necessary if you using hierarchical zones, but this option can be useful if you are migrating from classic to hierarchical zones.
5. Click **Enable auto-provisioning for group profiles**.
6. Click the Find icon to search for and select the “groups” zone provisioning group as the Source Group.

   If you followed the recommended naming convention, search for and select parentZoneName_Zone_Groups. For example, if the zone name is arcadeGlobal, select arcadeGlobal_Zone_Groups.
7. Select a method for assigning a new GID to new UNIX group profiles:

- **Generate from group SID** generates new GIDs that are guaranteed to be unique in the forest based on the Active Directory security identifier (SID) of the group. Selecting this option ensures groups defined in the parent zone have a unique GID across all zones in the Active Directory forest.
- **RFC 2307 attribute** uses the gidNumber attribute from the RFC 2307 schema to define GID values for the Active Directory groups that you add to the parent zone. This option requires you to add the RFC 2307 attribute to Active Directory group principals.
- **Use auto-incremented GID** selects the next available GID in the parent zone. In most cases, you should avoid using this option because it does not guarantee unique GIDs.
- **Generate using Apple scheme** generates group GIDs based on the Apple algorithm for generating numeric identifiers from the Active Directory group's objectGuid. This option is only supported for hierarchical zones.

8. Select a method for assigning a new group name to new UNIX group profiles:
   - **SamAccountName attribute** generates the group name for UNIX group profile based on the sAMAccountName value.
   - **CN attribute** uses the common name attribute to define group names for the Active Directory groups you add to the zone. You should only select this option if you verify the common name does not contain spaces or special characters. Otherwise, you should not use this option.
   - **RFC 2307 attribute** uses the cn attribute from the RFC 2307 schema to define group names for the Active Directory groups you add to the zone.
   - **Zone default value** uses the Group name setting from the Group Defaults tab to define group names for the Active Directory groups you add to the zone. In most cases, the default is a variable that uses the sAMAccountName attribute.

     By default, all UNIX group names are lowercase and invalid characters are replaced with underscores.
9. Click **OK** to save your changes.

Add Security Groups to the Parent Zone

The most common way to provision UNIX users is to use a private group identifier as the primary group. With this approach, every user has a unique primary GID that is the same as the UID.

Although not required, another common approach to provisioning UNIX users involves adding a small number of key security groups to the parent zone. For example, if you have a commonly-used group such as All US Employees to which you normally add valid Active Directory users as members, you could add that security group to the parent zone to assign all UNIX users the same primary GID in the parent zone. This approach makes provisioning UNIX users easier because you have already defined Active Directory users as members of that group. If you want to use an Active Directory group to set the primary GID for provisioned users, keep in mind that the size of the group membership can affect the performance of the Zone Provisioning Agent and how long it take to complete provisioning.

If you choose to have the user's primary group defined by Active Directory group membership, the Active Directory group must be in the same Active Directory forest as the users being provisioned. If the Active Directory group is located in another forest, provisioning fails.

If you want to use this approach:

1. Add the security group to the provisioning group for the parent zone (for example, parentZoneName_Zone_Groups).
2. Open the Properties for the parent zone, click the **Provisioning** tab, and define the business rules for the UNIX group profile provisioning associated with the security group.

   At the next update interval, the Zone Provisioning Agent adds a profile for the group to the zone. You can also run the zoneupdate command to add the profile without waiting until the next update interval. For example:

   zoneupdate *zoneName*
3. Click the **User Defaults** tab for the parent zone, select the ellipsis **<...>** option for the Primary Group and select the GID for the group profile that the Zone Provisioning Agent added to the zone.

Defining The Business Rules For New Users In The Parent Zone

In addition to the business rules for group profiles, you configure similar rules for new UNIX user profiles. When you use hierarchical zones, you typically want to configure these business rules for the parent zone so that the profile can be inherited in all child zones. Remember that the profile, by itself, does not provide any access to the computers in the child zones, and that you can override any inherited attributes in any zone or on individual computers.

> The business rules you define only affect new UNIX user and group profiles. The imported legacy data remains unchanged, and the Zone Provisioning Agent will not modify any attributes on the existing user and group profiles.

To Configure The Business Rules For User Profiles In The Parent Zone

1. Start Access Manager.
2. In the console tree, expand the **Zones** node.
3. Select the top-level parent zone, right-click, then click **Properties**.
4. Click the **Provisioning** tab.

   If you are defining business rules for a parent hierarchical zone and want to establish a “source zone” for profile attributes, click **Advanced**. You can then select the Source zone for any or all user and group profile attributes. If you select Source zone for any attribute on the Advanced Provisioning page, you can click **Browse** to search for and select the zone to use as the source zone. In most cases, selecting a source zone is not necessary if you using hierarchical zones, but this option can be useful if you are migrating from classic to hierarchical zones.
5. Click **Enable auto-provisioning for user profiles**.
6. Click the Find icon to search for and select the “users” zone provisioning group as the Source Group.

   If you followed the recommended naming convention, search for and select parentZoneName_Zone_Users. For example, if the parent zone name is arcadeGlobal, select arcadeGlobal_Zone_Users.

   This is the same group to which you added the Active Directory users associated with imported user profiles as described in Add existing users to the provisioning group for the parent zone.
7. Select a method for assigning a new UID to new UNIX user profiles:

- **Generate from user SID** generates new UIDs that are guaranteed to be unique in the forest based on the Active Directory security identifier (SID) of the user. Selecting this option ensures users defined in the parent zone have a unique UID across all zones in the Active Directory forest.
- **RFC 2307 attribute** uses the uidNumber attribute from the RFC 2307 schema to define UID values for the Active Directory users that you add to the zone. This option requires you to add the RFC 2307 attribute to Active Directory user principals. Otherwise, you should not use this option.
- **Use auto-incremented UID** uses the next available UID in the parent zone. In most cases, you should avoid using this option because it can create UID conflicts with users in other zones.
- **Use custom ID** enables you to use the employeeId, employeeNumber, or uidNumber attribute as the UID for new users. You should only select the employeeId or employeeNumber attribute if your organization already populates the employeeId or employeeNumber attribute with a unique value for each user account.
- **Generate using Apple scheme** generates user UIDs based on the Apple algorithm for generating numeric identifiers from the Active Directory user's objectGuid. This option is only supported for hierarchical zones.

8. Select a method for assigning a new UNIX user login name to new UNIX user profiles:
   - **SamAccountName attribute** generates the user login name for new UNIX users based on the sAMAccountName attribute.
   - **CN attribute** uses common name attribute for user names. You should only select this option if you verify the common name does not contain spaces or special characters. Otherwise, you should not use this option.
   - **RFC 2307 attribute** uses the uid attribute from the RFC 2307 schema to define user names for the Active Directory users that you add to the zone. This option requires you to add the RFC 2307 attribute to Active Directory user principals. Otherwise, you should not use this option.
   - **Zone default value** uses the setting from the User Defaults tab for the zone. In most cases, the default is a variable that uses the sAMAccountName attribute.
9. Select a method for assigning a new shell and home directory to new UNIX user profiles.
   - **RFC 2307 attribute** uses the loginShell attribute for the shell and the unixHomeDirectory attribute for home directory from RFC 2307 schema for the default shell and home directory
   - **Zone default value** uses the values you define on the User Defaults tab, which can include runtime variables for the shell and home directory.

     Runtime variables are populated with platform-specific values when a user tries to log on to a UNIX computer. For example, if a user logs on to a Linux computer with a profile that uses the runtime variable for the home directory, the home directory is /home/username. If the user logs on to a Solaris computer, the runtime variable becomes /export/home/username.
10. Select a method for assigning a primary group to new UNIX user profiles.
    - **RFC 2307 attribute** uses the gidNumber attribute from the RFC 2307 schema for primary group values. This option requires you to add the RFC 2307 attribute to Active Directory user principals. Otherwise, you should not use this option.

- **Zone default value** uses the values you define on the User Defaults tab. This setting enables you to use a specific group profile as the primary group for all UNIX users. If you don't change the default value for the primary group on the User Defaults tab, the default primary group is a private group.
- **Private group** uses the user's UID as the primary GID.
- **Active Directory group membership** uses the Active Directory group with the highest priority as the primary UNIX group. With this option, the Zone Provisioning Agent checks which groups a user belongs to and a prioritized list of groups you have defined. If you select this option, click the Configure icon to search for and select the Active Directory groups to include in the prioritized list. This option allows different users to have different primary GIDs in the same zone.
- **Generate using Apple scheme** generates the user's primary group identifier (GID) based on the Apple algorithm for generating numeric identifiers from the Active Directory objectGuid for the user's primary group. Note that the user's primary group must configured for the zone. If the primary group is not configured for the zone, an error will be logged in the Windows Event Log when the user is provisioned. This option is only supported for hierarchical zones.
- **Generate from group SID** generates new primary GIDs based on the user's Active Directory primary group using the Centrify algorithm for generating GIDs.

  If you select the Active Directory group membership option and a user isn't a member of any of the groups in the list of prioritized groups, the Zone Provisioning Agent will not create a UNIX user profile for the user, because it won't be able to determine the primary group. As noted in Add security groups to the parent zone, the most common approach is to have all users assigned the same primary GID in a zone.

11. Click **OK** to save your changes.

    By default, the GECOS field in new UNIX user profiles is populated using the displayName attribute for the user.

How Hierarchical Zones Affect Provisioning

Because hierarchical zones enable profile attributes to be inherited, defining the business rules for new users and groups in the parent zone enables the Zone Provisioning Agent to generate consistent profiles for all child zones.

When you define a UNIX profile for a group or a user in a parent zone, the attributes are automatically inherited by all child zones. For groups, inheritance makes the group GID and group name available in all child zones. For users, inheritance gives every user defined in the parent zone the potential to log on to every UNIX computer. You then use role assignments to control which computers users can actually access, and, once you begin defining custom roles, what they can do on those computers.

By default, all of the attributes in each new profile are inherited from the parent zone. You can then override any of the attributes as needed in each of the child zones or on individual computers on a case-by-case basis. This flexibility enables you to establish a consistent UID and GID namespace across all zones based on unique SID and sAMAccountName values, while granting exceptions to the specific cases where you need them.

For individual computers, UNIX user and group profiles are inherited from the zone the computer has joined. Typically, this is a child zone or the child of a child zone. You can manually override any attribute or set of attributes for individual computers. Any attributes you do not override are inherited from the zone and the business rules you defined for the Zone Provisioning Agent.

Adding New Users to a Provisioning Group and a Role Group

For new Active Directory users to be effective users of a zone, they must be added to the parent zone's "users" provisioning group and to a role group. You can add users to these groups manually using Active Directory Users and Computers or you can update your existing provisioning process for modifying the membership of Active Directory groups to add users to the appropriate groups. The key points to understand are:

- Users are added to a **provisioning group** so that the Zone Provisioning Agent creates a UNIX profile for them. A user must have a complete UNIX profile to be a valid user on UNIX computers. Centrify recommends creating the profile in the parent zone, but you can create the profile in any zone or on individual computers.
- Users are added to a **role group** so that they have a valid role assignment that allows them to log on or perform specific tasks. Initially, you only have two possible role assignments, listed or UNIX Login, but you are likely to create more.

Add The User to a Provisioning Group

Using Active Directory Users and Computers, scripts, or internal procedures, the basic workflow for a new user would be similar to this:

1. A new Active Directory user requests access to UNIX computers.
2. You add the user principal name to an Active Directory group principal. If you are adding the user to the parent zone, you add the user to the "users" provisioning group parentZoneName_Zone_Users.

   If you wanted to create the profile in a child zone instead of the parent zone, you would add the Active Directory user to the childZoneName_Zone_Users. If you use some other naming convention for the provisioning group, you would search for and select that group.
3. The Zone Provisioning Agent monitors this group and at the next interval (or ondemand) creates a UNIX profile for the user in the zone, based on the business rules you defined.

   
   If you remove a user from the Active Directory provisioning group, the Zone Provisioning Agent removes the UNIX user profile from the zone.
4. You notify the user that a new UNIX profile has been created with information about the login name and initial Active Directory password to use.

Add the User to a Role Group

Users must also have a role assignment for the zone where you want to grant access. A role assignment is required before the UNIX user profile is usable.

Using Active Directory Users and Computers, scripts, or internal procedures, the basic work flow for a new user would be similar to this:

1. A new Active Directory user requests access to UNIX computers.
2. You add the user principal name to the appropriate Active Directory group principal. If you want to allow the user to log on to computers in a child zone, you add the user to the Login role group childZoneName_Role_Login.

If the user should be recognized but not allowed to log on, you would add the Active Directory user to the childZoneName_Role_Listed. After you have created custom roles, you would search for and select groups based on the specific rights a user needs.

3. Run the Zone Provisioning Agent update command in preview mode to verify your changes. For example:

   zoneupdate /p *zoneName*

4. Check the results of the zoneupdate preview, then run the command without the preview option to execute the business rules for provisioning. For example:

   zoneupdate *zoneName*

Adding a New Unix Group Profile to All Zones

If you want to make a new UNIX group available to all zones, you should first create a new Active Directory group. In most cases, groups are not shared across multiple zones because of the potential for privilege escalation based on group membership. However, the steps for creating a UNIX profile that spans all zones or only the computers in a specific zone are similar.

Using Active Directory Users and Computers, scripts, or your existing provisioning process, the basic workflow for a new group would be similar to this:

1. Create a new Active Directory group for access to UNIX computers in the UNIX Groups organizational unit (ou=UNIX Groups, ou=UNIX).

   For example, if you are creating a new Active Directory group for the denali project team in the parent zone arcadeGlobal, use Active Directory Users and Computers to create a new group named arcadeGlobal_denali.

2. (Optional) Add users to the group if you know who to add.

   For example, if you are creating the group for a new project and you have a list of authorized users for that project, you can click the Members tab to add those Active Directory users to the new group. If those Active Directory users have a valid UNIX profile and role assignment in the zone, their secondary group membership is updated with the new group.

3. Add the new Active Directory group to the appropriate zone provisioning group. If you are adding the group to the parent zone, you add the user to the “groups” provisioning group parentZoneName_Zone_Groups.

   If you wanted to create the profile in a child zone instead of the parent zone, you would add the Active Directory group to the childZoneName_Zone_Groups. If you use some other naming convention for the provisioning group, you would search for and select that group.

4. Run the Zone Provisioning Agent update command in preview mode to verify your changes. For example:

   zoneupdate /p *zoneName*

5. Check the results of the zoneupdate preview, then run the command without the preview option to execute the business rules for provisioning. For example:

   zoneupdate *zoneName*

6. The Zone Provisioning Agent creates a UNIX profile for the group in the zone based on the business rules you defined.

If you remove an Active Directory group from the Active Directory provisioning group, the Zone Provisioning Agent removes the UNIX group profile from the zone.

Using the Zoneupdate Program for Controlled Automation

You can use the zoneupdate.exe program with command line options to provision profiles in controlled way, allowing you to verify that profiles and access rights are defined correctly for subsets of users or groups without affecting the production environment.

At a minimum, you must specify the zone name or canonical name for the zone to use the zoneupdate.exe program. The command line options are similar to the options available on the Provisioning tab when you display a zone's properties.

For example, to use the provisioning properties defined for a zone, you only need to specify the zone name at the command line:

zoneupdate default

If you use the canonical name for the zone, you specify the full path to the zone:

zoneupdate "centrify.com/program data/Centrify/zones/default"

You can override the default provisioning properties for a zone by specifying one or more of the following command line options.

Options are not case-sensitive. If you specify an option more than once, only the last value is used.

Use this option	To specify
/z:ZoneName or /SourceZone:ZoneName	The name of a source zone. If you do not specify a zone name and there's not a source zone defined in the zone's provisioning properties, you cannot use the zoneupdate command to copy user or group attributes from one zone to another. A source zone is required for classic zones. It is optional for parent hierarchical zones, but can be useful if you are migrating from classic to hierarchical zones.
/d:DomainName or /Domain:DomainName	The name of the domain to process. If you do not specify a domain name, the zoneupdate program processes the Active Directory domain to which the computer is joined.
/dc:DCName or /DomainController:DCName	The name of the target domain controller to connect. No option - This will use the default domain controller of target domain.

/uu:Option or /UserUid:Option	The method to use to set the user's numeric identifier (UID) value. You can specify any one of the following values: Auto to generate UIDs based on the Active Directory domain name and the RID of a user object. This setting is equivalent to selecting Generate from user SID in the Provisioning tab. AppleScheme to generate UIDs based on the Apple algorithm for generating numeric identifiers from the Active Directory user object's objectGuid. This setting is equivalent to selecting Generate using Apple scheme in the Provisioning tab. RFC2307 to use the uidNumber attribute in the Active Directory RFC2307 schema for the user's UID value. ZoneDefault to use the UID defined on the User Defaults tab for the zone. If there's no default value, the Next UID value for the zone is used. SourceZone to copy the UID from the same user in a specified source zone. EmployeeId to copy the UID from the employeeId attribute of the user object. EmployeeNumber to copy the UID from the employeeNumber attribute of the user object. uidNumber to copy the UID from the uidNumber attribute of the user object. If you don't use one of these values, you can set the UID to not have any value. For example: /uu:empty If you use this setting, users will have an incomplete profile in the zone.
/un:Option or /UserName:Option	The method to use to set the user's name. You can specify any one of the following values: sAMAccountName to use the Active Directory user's sAMAccountName attribute as the UNIX user name. CN to use the user's common name (CN) attribute as the UNIX user name. RFC2307 to use the uid attribute in the Active Directory RFC2307 schema as the UNIX user name. ZoneDefault to use the user name defined on the User Defaults tab for the zone. If there's no default value zone, the sAMAccountName is used. SourceZone to copy the user name from the same user in a specified source zone. If you don't use one of these values, you can set the user name to an explicit value. For example: /un:hunter
/us:Option or /UserShell:Option	The method to use to specify the user's default login shell. You can specify any one of the following values: RFC2307 to use the loginShell attribute in the Active Directory RFC2307 schema as the default shell. ZoneDefault to use the shell specified on the User Defaults tab for the zone. SourceZone to copy the shell defined for the user in a specified source zone. If you don't use one of these values, you can set the login shell using an explicit value. For example: /us:/bin/bash

/uh:Option or /UserHomeDirectory:Option	The method to use to specify the user's default home directory. You can specify any one of the following values: RFC2307 to use the unixHomeDirectory attribute in the Active Directory RFC2307 schema for a user as the default home directory. ZoneDefault to use the home directory specified on the User Defaults tab for the zone. SourceZone to copy the home directory defined for the user in a specified source zone. If you don't use one of these values, you can set the home directory to an explicit value. For example: /uh:/home/hunter
/ug:Option or /UserPrimaryGroup:Option	The method to use to specify the user's primary group identifier. You can specify any one of the following values: AppleScheme to generate the user's primary group identifier (GID) based on the Apple algorithm for generating numeric identifiers from the Active Directory objectGuid for the user's primary group. Note that the user's primary group must configured for the zone. If the primary group is not configured for the zone, an error will be logged in the Windows Event Log when the user is provisioned. This setting is equivalent to selecting **Generate using Apple scheme** in the Provisioning tab. PrimaryGroupSID to generate the user's primary group identifier (GID) based on the Centrify algorithm for generating numeric identifiers from the Active Directory security identifier of the user's primary group. PrivateGroup to set the user's primary GID value to be the same as the user's UID value. RFC2307 to use the gidNumber attribute f in the Active Directory RFC2307 schema as the primary group identifier or a user. ZoneDefault to use the primary group specified on the User Defaults tab in the zone. If there's no default value, zoneupdate.exe will stop provisioning the user. SourceZone to copy the primary group defined for the user in a specified source zone. example: /ug:empty If you use this setting, users will have an incomplete profile in the zone. GroupMembership to set the user's primary GID based on the user's Active Directory group membership. If a user is a member of the Active Directory groups ops-mgrs and ops-labs and one of those groups has a UNIX profile in the zone but not the other, the group with the UNIX profile in the zone will be used as the primary GID for the user. If both group have a UNIX profile in the zone, the one with higher priority will be used. You can set the priority for selecting the primary group to use in the Access Manager console. If the priority is the same, zoneupdate.exe will stop provisioning the user. If you don't use one of these values, you can set the primary GID to not have any value.
/uc:Option or /UserGecos:Option	The method to use to specify the user's GECOS field. You can specify any one of the following values: RFC2307 to use the gecos attribute in the Active Directory RFC2307 schema for a user. ZoneDefault to use the value defined for the GECOS field on the User Defaults tab for the zone. If you don't use one of these values, you can set the primary GID value to an explicit value. For example: /uc:Thompson, Hunter S.

/gg:Option or /GroupGid:Option	The method to use to set the group numeric identifier (GID) value. You can specify any one of the following values: Auto to generate the GID based on the Active Directory domain name and the RID of a group object. This setting is equivalent to selecting Generate from group SID in the Provisioning tab. AppleScheme to generate GIDs based on the Apple algorithm for generating numeric identifiers from the Active Directory group object's objectGuid. This setting is equivalent to selecting Generate using Apple scheme in the Provisioning tab. RFC2307 to use the gidNumber attribute in the Active Directory RFC2307 schema for the group GID value. ZoneDefault to use the GID defined on the Group Defaults tab for the zone. If there's no default value, the Next GID value for the zone is used. SourceZone to use the GID defined for the group in a specified source zone. If you don't use one of these values, you can set the GID to not have any value. For example: /gg:empty If you use this setting, groups will have an incomplete profile in the zone.
/gn:Option or /GroupName:Option	The method to use to set the group name. You can specify any one of the following values: samAccountName to use the Active Directory group samAccountName attribute as the UNIX group name. CN to use the group's common name (CN) attribute as the UNIX group name. RFC2307 to use the cn attribute in the Active Directory RFC2307 schema as the UNIX group name. ZoneDefault to use the group name defined on the Group Defaults tab for the zone. If there's no default value, the sAMAccountName is used. SourceZone to copy the group name from the group in a specified source zone. If you don't use one of these values, you can set the group name to an explicit value. For example: /gn:apps-lab
/u:ADGroupName or /UserSource:ADGroupName	An Active Directory group to use to populate a Centrify zone with users. Use the sAMAccountName and, optionally, the domain name to identify the group. For example, to use the Active Directory engineers group in the currently connected domain to populate users in the default zone: zoneupdate /u:engineers default To use the Active Directory engineers group in a specific domain, you can use the /d:DomainName option or group_name@domain_name. For example to use the Active Directory engineers group in the testdomain.org domain to populate users in the default zone: zoneupdate /u:engineers@testdomain.org default
/g:ADGroupName or /GroupSource:ADGroupName	An Active Directory group to use to populate a Centrify zone with groups. Use the sAMAccountName and, optionally, the domain name to identify the group. For example, to use the Active Directory employees group in the currently connected domain to populate groups in the default zone: zoneupdate /g:employees default

/v or /Verbose	Display detailed information about the provisioning of users and groups. When you use this option, the output format is: Group: groupname:gid User: uid:username:shell:home:primarygid
/p or /Preview	Preview the users or groups to be provisioned or removed. In preview mode, the zoneupdate.exe program does not create or remove any UNIX profiles.
/el or /EventLog: Level	Enable logging to the Event log. You can use the Event Viewer to check the log results. For the log level, you can specify any one of the following values: None - don't write any provisioning activities to the Event log. This is the default setting. Normal - Write only the name of the provisioned users and groups to the Event log. Verbose - Write the UNIX profiles for the provisioned users and groups to the Vent log.
/l or /Log:Level	Enable logging and set the level of detail recorded in the log file. For the log level, you can specify any one of the following values: Error to log only error messages. Warning to log warnings and error messages. Information to log informational messages, warnings, and errors. Verbose to log all messages, including details about the user and group profiles created or removed. Logging is off by default. If you enable logging, the default file location for the log file is: C:\Users\user_name\AppData\Roaming\Centrify\Zone Provisioning Agent\Log You can change the default log file path by modifying the following registry key: HKEY_LOCAL_MACHINE\Software\Centrify ZPA\LogLevel

Using Any Active Directory Attribute in a Profile

In addition to the provisioning properties you can set for a zone using Access Manager, you can manually configure the Zone Provisioning Agent to use any attribute in Active Directory to define a value for any field in automatically-provisioned UNIX user or group profiles. For example, if your organization uses a custom attribute, such as org_global_id, for all users, you can manually configure the Zone Provisioning Agent to use that attribute for the numeric user identifier (UID) in automatically-generated user profiles.

To manually specify an Active Directory attribute to use in a UNIX profile:

1. Open Microsoft ADSI Edit.
2. Select a target zone, right-click, then click **Properties**.
3. Select the **description** attribute, then click **Edit**.
4. Type a profile provisioning attribute and specify the Active Directory attribute to use for the profile.

   The valid provisioning attributes are:

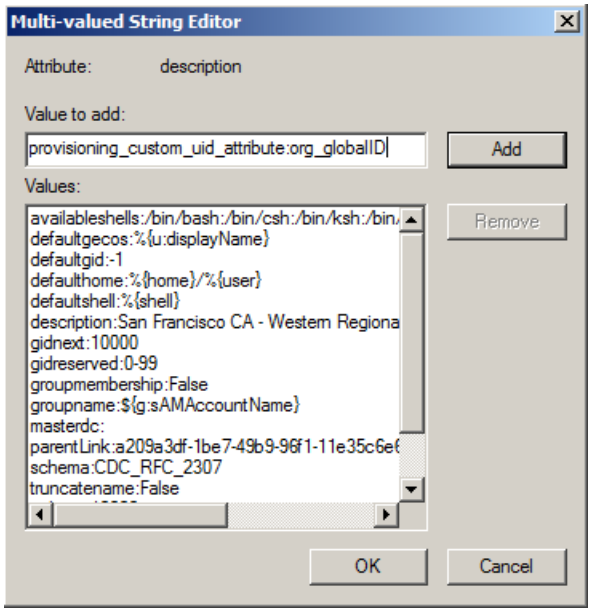
   provisioning_custom_uid_attribute
   provisioning_custom_gid_attribute
   provisioning_custom_primary_group_attribute
   provisioning_custom_user_unixname_attribute
   provisioning_custom_group_unixname_attribute

provisioning_custom_home_directory_attribute
provisioning_custom_shell_attribute

The format for the entry is:

provisioning_custom_uid_attribute:attribute_name

Replace *attribute_name* with the Active Directory attribute you want to use. For example:



5. Click **Add**, then click **OK**.
6. Run the Zone Provisioning Agent update command in preview mode to verify your settings. For example:

   zoneupdate /p *zoneName*
7. Check the results of the zoneupdate preview, then run the command without the preview option to execute the business rules for provisioning. For example:

   zoneupdate *zoneName*

If the Active Directory attribute type is different from the target profile value, the Zone Provisioning Agent attempts to convert the data type. If the data conversion fails, the Zone Provisioning Agent reports an error and stops the provisioning process.

Provisioning Users When Across Trusted Domains

The Zone Provisioning Agent includes two utilities in its Tools folder to assist you in provisioning users when there are trust relationships between domains. These two utilities are CopyGroup and CopyGroupNested. These utilities enable you to mirror group membership or a group hierarchy from a trusted domain and forest in a target domain and forest.

To use these command line utilities, you must have an account that can log on to the trusted source domain and the target domain. The account should also have read permission on the source domain and permission to update the target domain.

For example, assume you have configured the AJAX domain to have a one-way trust with the ACME domain and you have your Active Directory users and groups defined in the ACME domain. If you want to allow the users and groups in the ACME domain to log on to computers that are joined to the AJAX domain, you can log on to the AJAX domain controller with an account that has administrative privileges in both the AJAX and ACME domains, then run the CopyGroup utility to mirror the group membership from a group in the ACME source domain as zone users in the AJAX target domain.

For more information about the command line arguments and options for these utilities, see the usage message displayed for each utility.

Monitoring Provisioning Events

The Zone Provisioning Agent writes events to the Windows event log. You can use tools that monitor the event log to notify you of specific provisioning events. The following table lists the event identifiers and messages the Zone Provisioning Agent records.

Event	Severity	Event description and sample message
1	Information	Summarizes the result after a provisioning run. Centrify zones updated. Domain controller: domain_controller_name Container: container_name Start time: start_ time End time: end_time Successful: successful_message Failure: failure_message
2	Error	Indicates that provisioning failed for a specific domain because the domain was not found. Domain domain_name not found.
3	Error	Indicates that provisioning failed because the domain controller was not accessible. Domain server domain_controller_name is down.
4	Error	Indicates that provisioning failed because the domain was not operational. Domain domain_name is not operational.
5	Error	Indicates that provisioning failed without a specific cause. Failed to update zones in domain domain_name on domain controller domain_controller_name.
6	Error	Indicates that provisioning failed because there was a problem with the agent connecting to the Active Directory. Failed to update zones in domain domain_name on domain controller domain_controller_name. Error on Active Directory service.
7	Error	Indicates that provisioning failed because there was an unexpected error when updating the zones in a specific domain. Unexpected error when updating the zones in domain domain_name on domain controller domain_controller. Details: detail_ message.
8	Information	Provides verbose user provisioning information, including details about the provisioned user profile. User provisioned to a Centrify zone. User: user Zone: zone UID: uid Name: name Shell: shell Home directory: home_directory Primary group: primary_group Gecos: gecos

9	Information	Provides basic user provisioning information. User provisioned to a Centrify zone. User: user Zone: zone
10	Information	Provides verbose user de-provisioning information, including details about the user profile removed. User removed from a Centrify zone. User: user Zone: zone UID: uid Name: name Shell: shell Home directory: home_directory Primary group: primary_ group Gecos: gecos
11	Information	Provides basic user de-provisioning information. User removed from a Centrify zone. User: user Zone: zone
12	Information	Provides verbose group provisioning information, including details about the provisioned group profile. Group provisioned to a Centrify zone. Group: group Zone: zone GID: gid Name: name
13	Information	Provides basic group provisioning information. Group provisioned to a Centrify zone. Group: group Zone: zone
14	Information	Provides verbose group de-provisioning information, including details about the group profile removed. Group removed from a Centrify zone. Group: group Zone: zone GID: gid Name: name
15	Information	Provides basic group de-provisioning information. Group removed from a Centrify zone. Group: group Zone: zone
16	Warning	Indicates that the Zone Provisioning Agent received a warning message during the provisioning process. Warning occurred when provisioning a Centrify zone. Zone: zone Details: detail_message
17	Error	Indicates that provisioning failed because there was a problem with the permissions on the account used to run the Zone Provisioning Agent. Insufficient permission to setup the log file. Please contact your local administrator. Details: detail_message
18	Error	Indicates that provisioning failed because there was a problem with creating the log file in the log file location. Failed to create the log file. Please contact your local administrator. Details: detail_message
19	Information	Indicates that provisioning is paused to allow another provisioning cycle to complete. Zone Provisioning Agent failed to start another provisioning cycle at current_time because the previous provisioning cycle is not yet complete. The provisioning request is pending until Zone Provisioning Agent finishes the previous provisioning cycle.

20	Error	Indicates that provisioning failed because the computer was not found in the domain being provisioned. This computer is not joined to a domain or the domain is not available.
21	Error	Indicates that provisioning failed because there was a problem loading domain information. Failed to load domain information. No zone will be provisioned. Error: error_message
22	Error	Indicates that provisioning failed because there was a problem with the functional operation of the domain. Functional error occurred with domain domain_name.
23	Error	Indicates that provisioning failed because authentication failed for the account used to run the Zone Provisioning Agent. Domain domain_name authentication failed.
24	Error	Indicates that provisioning failed because the authentication system failed. Domain domain_name authentication (system) failed.
25	Error	Indicates that provisioning failed because there was an unexpected error when loading a specific domain. Unexpected error occurred when loading domain domain_name.
26	Error	Indicates that provisioning failed because the Active Directory forest was not found. Forest forest_name not found.
27	Error	Indicates that provisioning failed because the root domain controller was not accessible. Forest server server_name is down.
28	Error	Indicates that provisioning failed because the forest was not operational. Forest forest_name is not operational.
29	Error	Indicates that provisioning failed because there was a problem with the functional operation of the forest. Functional error occurred with forest forest_name.
30	Error	Indicates that provisioning failed because authentication failed at the forest level for the account used to run the Zone Provisioning Agent. Forest forest_name authentication failed.
31	Error	Indicates that provisioning failed because the authentication system failed at the forest level. Forest forest_name authentication (system) failed.
32	Error	Indicates that provisioning failed because there was an unexpected error at the forest level. Unexpected error occurred when loading forest forest_name.
33	Error	Indicates that provisioning failed because there was an error during the provisioning process. Error occurred when provisioning a Centrify zone. Zone: zone Details: detail_message

34	Warning	Indicates that provisioning might be incomplete because the account used to run the Zone Provisioning Agent does not have permission update zone in a specified domain. Insufficient permission to update the zones in domain domain.

Adding New Profiles Manually

Provisioning groups enable automated provisioning of UNIX profiles for users and groups with the Zone Provisioning Agent. Server Suite does not require you to use provisioning groups and business rules to create new users and groups. You can also manually create UNIX users and groups in any zone using the Access Manager console, ADedit, or custom scripts. Adding profiles manually to a zone provides you with control over the definition of individual UNIX profiles on a zone-by-zone basis.

Validating Operations After Deploying

This section provides sample activities for creating test cases and performing a formal validation of the Server Suite deployment. Although not required, executing a set of test cases that exercise Server Suite functionality will help you validate operations before extending the deployment to additional computers in the enterprise.

The specific use case scenarios and test cases you execute will depend on your organization's goals and requirements.

Understanding Testing as Part of Deployment

Most organizations initially deploy in a lab environment that simulates the production environment. The lab environment allows you to test the planned changes to system and user management processes. For example, if you plan to automate migration using scripts, you should build and test the operation of your tools to verify they work as you intend. After testing in a lab, most organizations move to a pilot deployment with a limited number of computers and users to continue verifying that authentication, authorization, and directory services are all handled properly in a real-world environment before moving to a full-scale migration across the enterprise.

In many cases, the pilot deployment also requires more formal testing of specific use cases to validate the deployment before rolling out software to additional computers. This phase of the deployment may also include activities designed to help users transition to Active Directory.

To validate the deployment:

- Execute test cases that verify authentication.
- Execute test cases the verify authorization rules for login access and restricted access.
- Execute test cases that verify the provisioning process.
- Execute test cases that address issues unique to your organization or your user community.

Other activities you may want to perform as part of the validation process include:

- Test configuration changes and customize configuration parameters.
- Document test results.
- Troubleshoot any authentication or authorization issues, if any are found.
- Train staff on new procedures.

- Communicate process changes to the users who are migrating to Active Directory.

  For example, if you plan to eliminate local account access, implement stricter password policies, or apply new access controls, you should prepare the user community for these changes.

Validating Basic Authentication and Password Policy Operations

Before you begin testing organization-specific scenarios, such as the integration of migration scripts or customized access control policies, you should verify basic operations are handled as expected. At a minimum, you should perform some of the following tests to verify basic operations:

- Verify a UNIX profile exists for the Active Directory users and groups to be used in testing.
- Verify the UNIX computer has successfully joined an Active Directory domain and the computer account has been created correctly.
- Verify an Active Directory user assigned the UNIX Login role is authenticated and can access computers in the zone used for testing.
- Verify migrated users assigned the UNIX Login role can log on using their UNIX or Active Directory user name and Active Directory password.
- Verify an Active Directory user assigned the Listed role has a valid UNIX profile, but cannot log on to computers in the zone used for testing.
- Verify that workstation authorization or account lockout policies defined in Active Directory are enforced.
- Verify that password management policies defined in Active Directory are properly enforced.
- Verify that a previously authenticated user is authenticated successfully when the UNIX lab computer is offline.
- Verify that common lookup commands and commands that require user and group information work as expected.

You can verify authentication, authorization, and password policy operations by setting options in Active Directory and attempting to log on and log off the computers in the pilot deployment.

Running Commands on the Unix Computer to Verify Operations

To confirm that a UNIX computer is a member of the Active Directory domain, you can run commands that retrieve information from Active Directory on the UNIX lab computer itself or by viewing the UNIX computer's account information in Active Directory Users and Computers or the Access Manager console.

Verify the Computer is Joined to Active Directory

To verify a computer is joined to the Active Directory domain and is retrieving information from Active Directory by running commands on the UNIX computer:

1. Log on to the UNIX computer.
2. Type the following command to retrieve information about the computer's connection to Active Directory:

   adinfo

This command returns basic information such as the host name for the computer, whether the computer is joined to the domain, and whether the computer is currently connected to Active Directory. For example:

Local host name: magnolia
Joined to domain: ajax.org
Joined as: magnolia.ajax.org
Current DC: ginger.ajax.org
Preferred site: Default-First-Site-Name
Zone: ajax.org/Centrify/Zones/default <!--TODO: company name in file> Last password set: 2017-12-21 11:37:22 PST
CentrifyDC mode: connected

For more detailed information about the environment, you can use --diag or other options with the command. For information about the options available and the information displayed for each option, see the adinfo man page.

3. Type the following command to verify that the adclient process is running:

   ps -aef|grep adclient

   The command should return output similar to the following:

   root 1585 1 0 14:50 ? 00:00:29 adclient

4. Type the following command to confirm that lookup requests use the information in Active Directory:

   getent passwd

   The command should list all of the Active Directory user accounts that are members of the zone and all local user accounts in the
   /etc/passwd file format. For example:

   ben:x:601:100:Ben Waters:/home/ben:/bin/bash
   ashish:x:501:100:Ashish Menendez:/home/ashish:/bin/bash
   sunni:x:900:100:Sunni Ashton:/home/sunni:/bin/bash
   jolie:x:502:100:Jolie Ames:/home/jolie:/bin/bash
   pierre:x:1001:100:Pierre Leroy:/home/pierre:/bin/bash

5. Review the contents of the /var/log/messages file and look for messages that indicate authentication problems or failures.

Verify Authentication for an Authorized User

To verify that an authorized Active Directory user can log on to a UNIX computer:

1. Restart the computer to display a logon screen or prompt.
2. Log on with the Active Directory user account you created for testing and provide the Active Directory password for that account.
   - The user account must have a complete UNIX profile.
   - The user account must be assigned the UNIX Login role.

- If the user account has been configured to set a new password at the next logon, you should be prompted to change the password. In this case, you must type and confirm the new password before continuing.

3. Log on using the UNIX login name for the user account and the Active Directory password.
4. Type commands to check the UID and GID assignments, home directory ownership, and other information for the logged on user.

Test Additional Administrative Tasks

You may want to try other typical administrative tasks that you expect to perform in the production environment. For example, you may want to test and verify the following tasks:

- Changing the password for an Active Directory user using the passwd or adpasswd command on a UNIX computer changes the user's Active Directory password for Windows computers.
- Logging on as a user from another trusted Active Directory domain or another trusted forest is successful when you specify the user's fully-qualified domain name (for example, milo.cutter@paris.arcade.com).
- Setting a user's effective group membership using the adsetgroups command.
- Logging on using previously cached credentials. Offline authentication enables users to log on when computers are disconnected from the network or have only periodic access to the Active Directory domain. For example, users who have laptop computers must be able to log on and be successfully authenticated when they are not connected to the network.

Resolving Issues in the Pilot Deployment

Executing a formal test plan is intended to help you uncover issues that need to be resolved, troubleshoot any unexpected behavior, and correct any potential problems before end-users are affected. The pilot deployment enables you to deploy Server Suite software packages on a subset of typical users in the production environment in a controlled way. You can then use the pilot deployment to evaluate server and network load and how adding new computers and users to the Active Directory affects your environment. The initial deployment also allows you to closely monitor the experience of the user community participating in the pilot program.

With the pilot deployment, you can also develop and refine your processes and operational expertise before you roll out Server Suite authentication and authorization services to the entire organization.

You can install Server Suite Agents at any time without affecting any user or computer operations. Installing the Server Suite Agent on UNIX computers has no effect until you join the computer to the domain. Setting up the initial zone or set of zones does not affect the operation of any existing Active Directory infrastructure or Windows environment. Therefore, you can install the software for the pilot whenever it is convenient to do so.

Before you join computers to the domain, you must define and assign appropriate roles to the user community. Users who don't have role assignments will not be allowed to log on to any computers. It is essential for you to test and validate role definitions and assignments to ensure users won't be locked out of the computers they need access to when computers join the domain.

Preparing Training and Documentation for Administrators and Users

The deployment team should develop and deliver training for end-users, technical support personnel, help desk operators, and account fulfillment personnel. This role-based internal training will help new team members come up to speed and also help with the resolution of technical issues.

You should also train staff members to understand that there will be two fulfillment processes in place during migration: the legacy account fulfillment process for computers that have not joined the domain and a new account fulfillment process for computers that have joined an Active Directory domain. Both fulfillment processes should be clearly documented and staff should be trained on how to determine which process to use. For example, training material should indicate how the UNIX provisioning team can determine whether a computer is in a zone, so that members know whether to use the legacy process or the new process.

After a computer is migrated to Active Directory, you should not allow any local account provisioning on that computer. You should also be sure that this is clearly documented in training materials, especially if you don't have centralized management of account creation policies. If you don't prevent local account provisioning, orphaned and noncompliant UNIX accounts can continue to exist, may create conflicts in the UID and GID namespace, and create audit compliance issues because they are not included in required reports.

As you migrate each set of computers to an appropriate zone, you should also notify all affected users before you complete the migration. This notification can take the form of an email, voicemail, meeting with project personnel or management, or any other logical combination. Notifying users in advance helps to reduce the number of account lockouts caused by UNIX users attempting to log on using their old UNIX password on migrated computers.

Deploying to the Production Environment

After the initial deployment is stable and you have migrated existing users and groups successfully, you can begin moving the rest of your UNIX computers, users, and groups to Active Directory. In most cases, this migration is done in stages by repeating the tasks described in this guide for additional target sets of computers, users, and groups. After each stage, you should allow a period of time for monitoring and resolving issues for the migrated user population. Your deployment plan should include a schedule for when different sets of users are to be migrated and an analysis of how those users should be placed into zones according to your migration plan.

In general, you should migrate an increasing number of computers into zones in each stage of the production deployment. For example, in the first round of migration, you might migrate 15% of the computers into the first set of zones. You should then allow time in the schedule to troubleshoot and resolve issues to ensure that the migration was successful. In the next phase, you then might migrate another 25% of the computers. After you determine the second phase of the migration is successful, you might migrate the remaining computers into the remaining zones.

Whenever possible, you should also plan to migrate all of the computers that have been identified for a particular zone at the same time. Migrating all of the computers in a zone at the same time helps to reduce user confusion over which password to use when authenticating.

Training the Support Staff for a Production Deployment

You should provide the following information or training to the IT support staff who are responsible for a set of users to be migrated to Active Directory:

- **Review of the deployment project plan**. Have the support staff read the deployment plan and review any changes to policies or procedures that deployment will entail.
- **Schedule for deployment**. Make sure members of the support staff are aware of when the deployment is scheduled to take place and that they will be available at that time and for a reasonable period thereafter.

- **Location of documentation**. Make sure all internal, operating system, and Server Suite-specific documentation is available so that support staff can use those documents to help them resolve any end-user issues that arise during the production deployment.
- **Pilot deployment experience and feedback**. Explain the result of the pilot deployment, including any issues encountered during the pilot and the resolution for each issue.
- **Common Windows and Active Directory administrative tasks**. For support staff members who are familiar only with supporting UNIX-based computers, provide training about Windows and Active Directory concepts and administration, as appropriate. If administrators will be using Windows-based programs or scripts to manage UNIX users and computers, they may need training specific to those tools. For example, administrators may need training to use Active Directory Users and Computers, Visual Basic scripts, or Access Manager console to manage Active Directory data.
- **Common UNIX administrative tasks**. For support staff members who are familiar only with supporting Windows-based computers, provide training about UNIX concepts and administration, as appropriate. If administrators will be using UNIX-based programs or scripts to manage UNIX users and computers, they may need training specific to those tools. For example, administrators may need training to create and use ADedit or LDAP scripts to manage Active Directory data.
- **Common access control and privilege management tasks**. Make sure members of the support staff are familiar with the tasks described in the *Administrator's Guide for Linux and UNIX*, and provide hands-on training in performing the most common of those tasks.
- **Internal policies and procedures specific to your network and business environment**. Create an operations handbook with details about common scenarios the support staff may be required to address, such as adding new UNIX computers or users to Active Directory.
- **Reporting and tracking issues related to Server Suite software**. Make sure support staff members know how to report issues or problems with authentication, authorization, or directory services. If your organization uses a bug or problem-ticket system for tracking issues, set up a new subject area for Server Suite-related issues.

Preparing the User Community in a Production Deployment

As you prepare to migrate a set of users to Active Directory, you should provide training or informational materials to inform that user community about what to expect. For example, if your organization has decided to implement policies that prevent locally-defined user accounts from accessing some computers, be sure that the user community affected by this policy understands the change. Similarly, if your organization has decided to eliminate service accounts or restrict access to computers previously available, you should communicate these changes and notify users about any migration issues that may affect file access permissions and file ownership.

When you are ready to migrate a specific set of users, you should inform the user population about the upcoming deployment by providing the following information:

- **Schedule for deployment**. Make sure that department managers and end-users know when the switch to Active Directory is scheduled to occur.
- **Computers and applications affected**. Make sure that department managers and end-users know if their workstations or the servers they access for business applications are included in the deployment. If users need access to a computer that is being added to an Active Directory domain, they need to know whether their user account is in the same domain as the computer or a different domain. If there are applications hosted on a computer that is being added to an Active Directory domain, users need to know how this will affect access to

the hosted application. For example, users may need to select a domain when logging on, or log on using the user_name@domain_name format.

- **Active Directory account information**. Make sure that end-users know their Active Directory account information and understand that they must use their Active Directory password to access their UNIX workstations after the deployment is complete. You should inform users about the valid logon names and formats they can use, the Active Directory password assigned to their account if it is a new account, whether they are required to change their password when they next log on, and any password complexity rules you have implemented. Active Directory may lock accounts if users attempt to log on using their UNIX password, which could result in a large number of Help Desk requests for password resets.
- **Changes to access policies**. Make sure that department managers and end-users are aware of any changes to access control policies. For example, if you are using group policies to deny access to some users or groups who could previously log on to a computer, you should inform those users or groups of the change and that it will take effect after the migration to Active Directory.

Populating Zones in a Production Environment

In planning your deployment, you should have determined your basic zone requirements and how you will migrate existing user communities to Active Directory. Based on your analysis, you should have a zone design with one or more parent zones and the child zones for each parent to define a candidate set of users and groups with the potential to access a given set of computers.

Typically, you should focus on one zone at a time, importing and mapping the existing users to Active Directory accounts. You should also determine whether you need to create new Active Directory accounts for any of the existing users or groups you are importing. If possible, you should use Active Directory group membership and role assignments to manage access for UNIX users and groups, you import.

Joining a Domain in a Production Environment

In smaller organizations or organizations where individual users have permission to join their own workstation to the Active Directory domain, you can run the adjoin command interactively on individual computers. This option works well when computers are distributed across many different domains or when individual users are joining their own workstation to the domain.

In larger organizations, however, you may want to use a custom script to remotely join a group of UNIX computers to an Active Directory domain. If you develop a custom script for joining a domain, the script should restart services or reboot the computers where it runs.

After joining a domain, you should monitor computers closely for a few days before extending the deployment to additional computers.

If the join operation fails or users cannot log on, you can run the adleave command to restore the computer to its previous state.

Defining Role-Based Access for Users and Computers

By default, Server Suite includes two roles—the listed role and the UNIX Login role—that are required for migration. This section discusses additional role-based controls you can define for better management of privileged access and authorized activity.

If you have well-defined access rules and command privileges in sudoers configuration files, you can import those definitions and use them as the basis for creating custom roles in Access Manager. For information about importing sudoers files and converting the imported definitions into roles, see the *Administrator's Guide for Linux and UNIX*.

Addressing the Business Problem of Role-based Security

Privilege management and role-based access controls are approaches to the basic business problem of securing an enterprise's key computer systems and sensitive data. Restricting access based on a user's role or specific job requirements can require you to make some difficult decisions about who has access to what and why access is granted or denied. These decisions also have the potential to disrupt user activity or existing business processes. Therefore, you should do thorough planning to identify the roles to implement, who should have permission to execute privileged commands, and who should have restricted access.

Defining the appropriate rights for users in different roles often requires negotiation with different groups in the organization to achieve the right balance of security and functional capability. Before implementing a solution, you should have these conversations and set expectations about what will change in the user's environment.

Creating a Root-Equivalent Role Definition

One of the first roles you should plan to create is an administrative role that is equivalent to specifying ALL:ALL in a sudoers file or giving users access to the root password on their computers. The purpose of this role definition is to allow selected users to execute privileged commands on a regular basis. The role definition allows them to execute commands without being given the root password or having privileges hard-coded in individual sudoers files on multiple computers.

Because this role definition enables system administrators to execute privileged commands without the root password, you can improve security for the organization and reduce the chance of an audit finding for access to the root password.

You can create this role definition in a parent zone or a child zone to control its scope. In most cases, you should only assign the role in a child zone or on an individual computers.

Define the Right for Running All Commands

Rights and roles are defined at the zone level and inherited down the zone hierarchy. If you define a right in the top-level zone, it is available in all child zones. If you define a right in a child zone, it can be used in that zone and any of its child zones. Similarly, you can define roles in the top-level parent or any child zone, depending on where you want to make the role available. In this example, the right to run all commands as the root user is defined in a top-level parent zone.

The following instructions illustrate how to define a right for running all commands using Access Manager. Examples of scripts that use the Access Module for Windows PowerShell, ADEdit, or the Server Suite Windows API are available in other guides, the *Server Suite Software Developer's Kit,* or in community forums on the Delinea website.

To define a right for running all commands as root:

1. Open Access Manager.
2. Expand Zones and select the top-level parent zone.
3. Expand **Authorization > UNIX Right Definitions**.
4. Select **Commands**, right-click, then click **New Command**.
5. On the General tab, type a name for this command right and, optionally, a description for this right, then define the right to run all commands like this:
   - Type an asterisk (*) in the Command field to indicate all commands are allowed.
   - Select **Specific path** and type an asterisk (*) in the field to indicate that any path is allowed.
6. Click the Restricted Shell tab and deselect the **Can be used in a restricted role** option if you want to prevent this command from being used in a role that uses a restricted shell environment.
7. Click the Run As tab to verify the command can be used by dzdo and is set to run as root by default.
8. Click **OK** to use the default environment variable settings and command attributes.

   Alternatively, you can click the Environment and Attributes tabs if you want to view or set additional properties for this right definition.

Create a Role Definition for Running All Commands

After you have defined the right to allow a user to run any command with root privileges, you can create a role definition for that right. You must create a role definition somewhere in the zone hierarchy before you can assign users to the role.

To create a role definition with the right to run all commands as root:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the role definition.
3. Expand Authorization.
4. Select **Role Definitions**, right-click, then click **Add Role**.
5. Type a name and description for the new role, then click **OK**.

   For example, type a name such as root_equivalent and descriptive text such as Users with this role can run any command with root privileges.

   Optionally, you can select **Allow local accounts to be assigned to this role** if you want to assign both Active Directory users and local users to the role. This option is only available when you first create a role definition. You can also click **Available Times** if you want to limit when the role is available for use. By default, roles are available at all times.

   If you are using the UNIX Login role to grant access to computers in the zone and want to use the default auditing level of **Audit if possible**, you can click **OK** then skip to Step 8.
6. If you are not assigning the UNIX Login role to grant access to computers, click the System Rights tab and select the following options:

- Password login and non-password (SSO) login are allowed
- Non-password (SSO) login is allowed
- Login with non-Restricted Shell

  Note that you cannot set these system rights if you selected the option to allow local users to be assigned to this role.

7. If you don't want to use the default auditing level, click the Audit tab.
   - Select **Audit not requested/required** if you have the auditing service enabled but don't want to audit user activity when this role is used.
   - Select **Audit if possible** to audit user activity where you have the auditing service enabled.
   - Select **Audit required** to always audit user activity. If the auditing service is not installed or not available, users in this role are not allowed to log on.
8. Select the new role definition, right-click, then click **Add Right**.
9. Select the right you defined for running all commands as root, then click **OK**.

Assign an Active Directory Group to the Role

As discussed in previous chapters, you should associate Centrify role definitions with Active Directory security groups so that you can manage them using the processes and procedures you have for managing Active Directory group membership. If you are using the recommended deployment structure and naming conventions, you would create a new Active Directory group in the ou=User Roles, ou=Centrify organizational unit using the format ZoneName_Role_RoleName. For example, you would create an Active Directory group named sanfrancisco_role_rootequivalent. You can then assign the new role definition to that group.

To assign the role definition to an Active Directory group:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to assign the role definition.
3. Expand Authorization.
4. Select **Role Assignments**, right-click, then click **Assign Role**.
5. Select the role definition you created for root-level access, such as root_equivalent, then click **OK**.
6. Click **Add AD Account** to search for and select the Active Directory security group you created for the role.
   - Select **Group** as the object to find.
   - Optionally, type all or part of the group name.
   - Click **Find Now**.

     Select the group you created for the role in the results, then click **OK**.
7. Click **OK** to complete the assignment.
8. Add members to the Active Directory security group for the role definition using Active Directory Users and Computers, an internal script, or another tool.

9. Test the role assignment by checking whether the user you added to the Active Directory group can execute privileged commands using dzdo in place of sudo.

   dzdo /usr/share/centrifydc/din/adflush

   Details about commands that are executed with dzdo are logged to the secure syslog facility on the computer where they were executed.

Creating a Restricted Role for a Shared Service Account

The root-equivalent role definition provides centralized management for a limited number of administrators who have permission to execute all commands on selected computers. Another common reason for defining a role is to execute privileged commands associated with a service account. In many organizations, service account passwords are known by multiple users, making them a security risk. For example, all of the database administrators in the organization might know the password for an oracle service account, an account with permission to perform privileged database operations. Because the password is shared information, it presents a security risk and a potential audit finding that might have costly consequences.

Setting up a role definition for a service account involves creating a command right for switching to the service account user and defining a PAM access right for role.

Define the Right for Switching to a Service Account

The steps for defining a right for switching to the service account user are similar to defining the rights for the root-equivalent user, but the definition is more restrictive.

To define a right for switching to a service account:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new command right.
3. Expand **Authorization > UNIX Right Definitions**.
4. Select **Commands**, right-click, then click **New Command**.
5. On the General tab, type a name for this command right and, optionally, a description for this right, then define the right to switch to the service account. For example, if the service account is oracle:
   - Type su - oracle in the Command field.
   - Verify the Standard user path is selected.
6. Click the Restricted Shell tab, under Can be used in a restricted role, select **Specific user or uid**, then type root.
7. Click the Run As tab, deselect **Can be used by dzdo**.

   These settings specify that this right can only be used in a restricted shell environment and users can only run the commands that are explicitly allowed in the restricted role they are assigned. If this is the only right defined for a role, the only command users assigned to the role can run is su - oracle. For a role definition with this right to be effective, you would add command rights for the specific database operations users should be allowed to perform after switching to the oracle service account. For example, if the oracle service account is used to run a backup-all-dbs script, you would add a right to allow the execution of that script.
8. Click **OK** to use the default environment variable settings and command attributes.

Alternatively, you can click the Environment and Attributes tabs if you want to view or set additional properties for this right definition.

Define a PAM Access Right to Allow Logging On

The default UNIX Login role allows users to log on using a password or without a password in an unrestricted environment. If you are creating a role definition for a service account, you can use PAM access rights to control the specific PAM-enabled applications users can use to log on. To illustrate controlling how users log on, this example of a restricted role for the oracle service account only allows users to log on with ssh.

To define a PAM access right for a specific application:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new PAM right.
3. Expand **Authorization > UNIX Right Definitions**.
4. Select **PAM Access**, right-click, then click **Add PAM Access Right**.
5. Type a name and, optionally, a description of the PAM application for which you are adding an access right.

   For the Application field, type the platform-specific name for the PAM application as defined in the PAM configuration file or PAM directory. For example, type ssh or sshd. You can also use wildcards in this field to perform pattern matching for the application name.
6. Click **OK** to save the access right for this PAM-enabled application.

Create a Restricted Role Definition for the Service Account

After you have defined the rights that allow a user to log on using a PAM-enabled application and run the su - command for a service account, you can create a role definition for these rights. You must create a role definition somewhere in the zone hierarchy before you can assign users to the role.

To create a restricted role definition for switching to a shared service account:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new role definition.
3. Expand Authorization.
4. Select **Role Definitions**, right-click, then click **Add Role**.
5. Type a name and description for the new role, then click **OK**.

   For example, type a name such as oracle_service and descriptive text such as Users with this role can start a secure shell session and switch to oracle.

   By default, this role is available at all times. You can click **Available Times** if you want to specify days of the week or select times of the day for making the role available.
6. Click the System Rights tab and select at least one option that allow users assigned to this role definition to log on, then click **OK**.

In this example, users open a secure shell to switch to the service account so you might select **Non-password (SSO) login is allowed**.

If a service account instead of a user account is used to log on, it might be mapped to a disabled Active Directory account. In this case, you might select the **Account disabled in AD can be used by sudo, cron etc** system right to ignore the disabled state and allow the service account to log on.

7. Select the new role definition, right-click, then click **Add Right**.
8. Select the rights you defined for running the switch user (su -) command and logging on with the PAM application ssh, then click **OK**.

Assign an Active Directory Group to the Role

As discussed in previous chapters, you should associate Server Suite role definitions with Active Directory security groups so that you can manage them using the processes and procedures you have for managing Active Directory group membership.

If you are using the recommended deployment structure and naming conventions, you would create the Active Directory group in the ou=Service Accounts, ou=Centrify organizational unit using the format ZoneName_Service_RoleName. For example, create an Active Directory group named sanfrancisco_service_oracle. You can then assign the new role definition to that group.

To assign the role definition to an Active Directory group:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to assign the role definition.
3. Expand Authorization.
4. Select **Role Assignments**, right-click, then click **Assign Role**.
5. Select the role definition you created for using secure shell and switching to the service account access, such as oracle_service, then click **OK**.
6. Click **Add AD Account** to search for and select the Active Directory security group you created for the role definition.
   - Select Group as the object to find.
   - Optionally, type all or part of the group name.
   - Click Find Now.

     Select the group you created for the role in the results, then click **OK**.
7. Click **OK** to complete the assignment.

Working in a Restricted Shell Environment

When users who are assigned to this role want to open a secure shell session and switch to the oracle service account, they will be placed in a restricted shell environment. Within the restricted shell, they can only execute the commands you have added to the role definition until they exit the restricted shell session. In this example, the role definition only allows users to log on using ssh and execute one command, su - oracle. If those users are also

assigned the UNIX Login role, they will have access to an unrestricted shell when they close the restricted shell session.

If you want users who access a shared service account to work exclusively within the restricted shell environment, you must remove the UNIX Login role assignment in the zone or on the computer where they should only have restricted shell access. Before removing the UNIX Login role assignment, however, you should consider the trade-off between improved operational security and audit compliance and reduced operational access. Depending on the rights you add to a role that runs in a restricted shell environment, the restricted shell can dramatically limit what users can do.

Testing Access in a Restricted Shell

If you create a role definition for a shared service account that runs in a restricted shell environment, you should test it before migrating any users to it. You can use the dzinfo command with the --test option from a UNIX command prompt. For example, type dzinfo, the user name to test, the --test option, then the full path to the command to test:

dzinfo raejames --test “/usr/bin/su - oracle

You can also run the dzinfo command with the --roles option to see information about the rights defined for the current user or a specified user. For example, run the following command to check the roles and rights defined for the user raejames:

dzinfo raejames --roles

For more information about using this command, see the dzinfo man page.

What Users See in a Restricted Shell Environment

For users assigned to a role that runs in a restricted shell, logging on opens a dzsh shell. Within that shell users can only execute the commands you have explicitly defined for them. In this example scenario for a shared service account, typing su - oracle is the only allowed command. If the user types any other command, the shell reports that the command is not allowed.

Creating a Role Definition for Temporary Root Access

Another common use case for role definitions occurs when you want to provide temporary access to privileged commands. For example, you might want to provide temporary rootlevel access to an application developer troubleshooting a problem on a production server or to a consultant you’ve hired for a specific period of time. These types of role definitions are often used as overrides on individual computers.

The steps for creating a role definition with temporary root access are similar to the steps for creating the other roles, except that you specify time constraints for the role. The time constraints might include specific hours of the day, days of the week, or a start and end time for a role assignment. The next sections summarize the steps for creating a role with temporary root-level access.

Define a Command that Allows Root Access

The steps for defining a right for switching to the root user are similar to defining the right to run commands for the root-equivalent user, but it is recommended that you create a separate right definition for this case.

To create the right to switch to the root user:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new command right.
3. Expand Authorization > UNIX Right Definitions.
4. Select **Commands**, right-click, then click **New Command**.
5. On the General tab, type a name, such as emergency_access, for this command right and, optionally, a description for this right, then define the right to switch to the root user:
   - Type the command for switching to the root user. For example, type su - root in the Command field.
   - Verify Standard user path is selected.
6. Click the Restricted Shell tab and verify **Can be used in a restricted role** and **User running the command are selected**.

   These options enable you to use this command right in combination with other rights in a role definition that requires a restricted shell environment.
7. Click the Run As tab and verify **Can be used by dzdo** and **Any user** are selected, then click **OK**.

   In most cases, you can leave the default settings for the other properties. If you want to make changes, click the Environment and Attributes tabs before saving the new command.

Create a Role Definition for Temporarily Running as Root

After you have defined the right to switch to the root user, you can create a role definition for that right.

To create a role definition with the right to run the emergency_access command:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new role definition.
3. Expand Authorization.
4. Select **Role Definitions**, right-click, then click **Add Role**.
5. Type a name and description for the new role.

   For example, type a name such as emergency_access and descriptive text such as Users with this role can temporarily run commands with root privileges.
6. Click **Available Times** to specify days of the week or select times of the day for making the role definition available.

   For example, you might want to allow access only on Friday, Saturday, and Sunday and deny access the rest of the week. After you have set the days and times for the role definition to be available, click **OK**.
7. Click **OK** to save the role definition.
8. Select the new role definition, right-click, then click **Add Right**.
9. Select the emergency_access command you defined for switching to the root user, then click **OK**.

   To use this role, a user must be assigned to the UNIX Login role for the zone or a role definition that has, at a minimum, the following System Rights:

- Password login and non-password (SSO) login are allowed
- Login with non-Restricted Shell

Assign the Role as a Computer-level Override

In most cases, a role definition of this type is assigned to a specific computer rather than applied to all computers in a zone.

To make a role assignment on an individual computer:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name that contains the computer for which you want to define a computer-level role assignment.
3. Expand Computers, then select the specific computer on which you want to make a role assignment.
4. Select **Role Assignments**, right-click, then click **Assign Role**.
5. Select the role definition you created for temporary root access, such as emergency_access, then click **OK**.
6. Click **Add AD Account** to search for and select the Active Directory user who should have temporary root access:
   - Leave User as the object to find.
   - Optionally, type all or part of the use name.
   - Click Find Now.

     Select the user in the results, then click **OK**.
7. Deselect **Start immediately** and set a specific Start time for the role assignment.
8. Deselect **Never expire** and set a specific End time for the role assignment.
9. Click **OK**.

Verify the Role Assignment on the Computer

You can run dzinfo --roles or dzinfo username --roles to see if the emergency_access role is available based on the start time for the role definition and the local time of the Linux or UNIX computer.

At the specified start time for the role assignment on the local computer, the user you assigned to the emergency_ access role can type the following command:

dzdo su - root

The user is not prompted to provide the password and becomes the root user on the local computer until the specified role assignment end time. The one caveat to be aware of is that the user would continue to have root access after the specified end time if the shell session remains open continuously. If a user is still logged on after the time period has expired, you should check whether the user still requires root-level access. If the session has remained open but the user should no longer have root access, kill the session and log the user off.

Creating a Role Definition With Specific Privileges

The previous examples of role definitions granted broad privileges. You can also use role definitions to grant or deny very specific rights. For example, you might want to deny access to a specific set of commands for a specific

group of administrators who otherwise have broad access rights or to strictly limit exactly what commands users can execute. Depending on the requirements of your organization, you might configure these types of role definitions to be used in a restricted or unrestricted shell.

The steps for creating a role definition with specific privileges are similar to the steps for creating the other roles. In this example, rights are defined to prevent the execution of specific commands and combined with a right to grant access to all commands not explicitly listed.

Define Command Rights to Prevent the Use of Commands

The steps for defining rights that deny access to specific commands are similar to the steps defining other rights, but require different syntax. In this example, you create a “blacklist” of commands users cannot execute.

To create the right to switch to the root user:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new command right.
3. Expand Authorization > UNIX Right Definitions.
4. Select **Commands**, right-click, then click **New Command**.
5. On the General tab, type a name, such as No password resets, for this command right and, optionally, a description for this right, then define the right:
   - Type !passwd * in the Command field.
   - Verify Standard user path is selected.

     An exclamation point (!) at the start of a command disallows matching commands. Command rights that start with the exclamation point take precedence over others that don’t.
6. Click the Restricted Shell tab and verify **Can be used in a restricted role** and **User running the command are selected**.

   These options enable you to use this command right in combination with other rights in a role definition that requires a restricted shell environment.
7. Click the Run As tab and verify **Can be used by dzdo** and **Any user** are selected, then click **OK**.

   In most cases, you can leave the default settings for the other properties. If you want to make changes, click the Environment and Attributes tabs before saving the new command.
8. Repeat Step 4 to Step 7to create rights for the following specific commands:

   !groupadd *
   !useradd *
   !groupdel *
   !userdel *

Create a Restricted Shell Role Definition that Uses the Command Rights

After you have defined all of the command rights that disallow specific commands, you can create one or more role definitions to use those rights. For example, you might create one role definition to run in an unrestricted shell that requires users to invoke dzdo to execute privileged commands and another role definition that runs in a restricted

shell but does not require users to execute privileged commands using dzdo. The second role might be useful if you have existing scripts that would have to be modified if invoking dzdo is required.

To create a role definition for specific command rights:

1. Open Access Manager.
2. Expand Zones and the individual parent or child zones required to select the zone name where you want to create the new role definition.
3. Expand Authorization.
4. Select **Role Definitions**, right-click, then click **Add Role**.
5. Type a name and description for the new role.

   For example, type a name such as operators and descriptive text such as Users with this role can run privileged commands but not reset passwords, add or delete users and groups.
6. Click **System Rights** if you want this role definition to be used in a restricted shell environment as a replacement for the predefined UNIX Login role.

   To use this role, a user must be assigned to a role definition that has at least one UNIX system right, such as Password login and nonpassword (SSO) login are allowed or Nonpassword (SSO) login is allowed.
7. Click **OK** to save the role definition.
8. Select the new role definition, right-click, then click **Add Right**.
9. Select all of the command right that disallow specific operations, the command right that grants access to all remaining commands, and a PAM access right, then click **OK**.

   For example, you might add the following previously-defined command rights to this role definition:

   No password resets
   No user adds
   No group adds
   No user deletes
   No group deletes
   Root like access (* for all commands not explicitly disallowed)
   PAM ssh/login allowed

   This role definition allows members of the operators role to execute any command within a restricted shell environment except those explicitly disallowed, including privileged commands, without invoking dzdo first. You can assign the role definition to the appropriate Active Directory users or groups like the previous role definitions.

Create an Unrestricted Shell Role Definition that Uses the Command Rights

The command rights were configured to allow execution in either a restricted shell environment or an unrestricted shell environment. In an unrestricted shell environment—for example, the default shell environment when users are assigned the UNIX Login role—commands that require administrative privileges must be executed by first invoking the dzdo command, which is similar to invoking commands with sudo.

You can control whether users are required to enter a password when they execute privileged commands using dzdo by setting the **Authentication required** on the Attributes tab when you create a command right. By default, no

password is required. If you were adding a new command right that requires authentication, you would click the Attributes tab, select **Authentication required** then select one of these options:

- **User's password** if users are required to enter their own password before executing the command.
- **Run as target's password** if users are required to enter the password for the target account that is executing the command.

In most cases, the default of no password is appropriate because the user has been previous authenticated before invoking dzdo to execute a privileged command and the **Run as target's password** option requires the user to know the privileged account password. For example, if the run-as user is root, the **Run as target's password** authentication option requires the user to know the password for the root account.

The steps for creating the role definition that includes the previously-defined command right are the same for the unrestricted shell as for the restricted shell except that at Step 6 of Create a restricted role definition for the service account in the System Rights tab, you would also select the **Login with non-Restricted Shell** option if you are not using the UNIX Login role. You could add all of the same command rights to the role definition and grant the same privileges and exceptions.

The primary difference between the two role definitions would be how users execute their privileged commands.

In the restricted shell environment, users running the adflush command requiring administrative privileges:

dzsh $ adflush

In the unrestricted shell environment, users running the adflush command requiring administrative privileges:

[tulo@ajax]$ dzdo adflush

Creating a Role Definition with Rescue Rights

The Rescue rights option allows you to control which users should be able to log on if problems with the authorization cache or auditing service are preventing all other users from logging on. For example, if you have a computer with sensitive information, such as credit card numbers or intellectual property, you might require auditing for all users in the role with access that computer. If the auditing service is stopped or removed on that computer, no one would be able to log on and use the computer until auditing is restored. If you create a role with the Rescue rights option selected, only the users assigned to that role are able to log on and continue working until the problem that caused the lockout is found and fixed.

Users who are in a role granted access because they have rescue rights can still be audited through the system logging facility. However, their activity is not recorded in the audit store database if the auditing service is not available.

Creating Additional Custom Roles and Role Assignments

The previous sections described common roles that organizations implement to begin the process of migrating and removing locally defined privileged accounts. For most organizations, locallydefined accounts with privileged access present a security risk and are often identified as a compliance issue by auditors.

By creating role definitions similar to those described in this chapter, you can eliminate the need to share root and service account passwords while still providing privileged access to computers where it's needed. These additional roles are not required, however. You can choose to create them or create a completely different set of role definitions to suit your organization. For example, you might decide to create custom roles specifically tailored to the needs of database administrators, backup operators, and web application developers. Similarly, you might

decide to create separate role definitions that are customized with AIX command rights for AIX administrators that are different from the command rights defined for Solaris administrators.

As with the common role definitions, additional custom role definitions can be created in the top-level parent zone and available throughout the zone hierarchy or in any child zone. They can also span all the computers in a zone or be assigned specifically to individual computers.

If you plan to create your own custom role definitions and role assignments, keep the following key points in mind:

- Rights associated with roles are cumulative. Users receive all of the rights in all of the roles they are assigned.
- Users must be assigned at least one role that allows an interactive login or Kerberos authentication to have any access to any computers. For existing users, this is accomplished by assigning the default UNIX Login role during the migration to Active Directory.
- Users must be given the Login with non-Restricted Shell system right to have access to a full shell. If they assigned in a role without this right, they can only execute the commands explicitly defined for their role.

For users who have previously had full shell access, this limitation can be frustrating, unexpected, and unworkable. Before placing or moving users into a restricted role, be sure those users and managers throughout the organization are well-informed and wellprepared for the change and understand the business reasons for the change.

Working with Computer Roles

In addition to the role definitions that confer specific rights when assigned to users and groups, Server Suite provides a mechanism for linking a specific group of computers to a group of users with a specific role assignment. These computer-based access rules, called **computer roles**, identify computers that share a specific attribute that you define and a set of users with common access rights.

For example, you can define a computer role that identifies a set of computers as Oracle database servers linked to a set of users who have been assigned the oracle_dba role. You can then add and remove users from the Active Directory role group linked to the oracle_dba role to grant or remove the rights associated with the oracle_dba role. In this example, the computer role identifies computers that host Oracle databases and the set of users assigned the database administrator role.

The same set of computers might include computers with AIX and Solaris operating systems. You could then create separate computer roles that link the AIX computers to a group of AIX administrators and the Solaris computers to a group of Solaris administrators.

Planning to Use Computer Roles

Because computer roles provide you with a great deal of flexibility for defining access rights, you might want to do some planning before you create new computer roles. For example, before you create a computer role you must know the criteria you want to use to group computers into one or more Active Directory security groups. You must also identify the users who will have a common set of access rights based on the computer grouping.

At a high-level, defining a computer role requires the following:

- Identify computer roles you want to define.

  Decide on the attribute the computers in a particular group share. For example, you can use a computer role to identify computers in the web farm, that host specific applications, or serve a specific department.

- Identify the users for the computer role and create Active Directory groups for them.

  You might need multiple groups because different sets of users have different access requirements. For example, if you are creating a computer role for a set of Oracle servers, you might need separate Active Directory groups for database users, database administrators, and backup operators.

- Identify the role definitions each set of users should be assigned.

  You might need to create specific access rights and role definitions for different sets of users. For example, if you are creating access rights for database users, database administrators, and backup operators, the database users may be able to use the predefined UNIX Login role, while administrators need permission to run privileged commands, and backup operators might be assigned a limited set of commands in a restricted shell.

How Computer Roles Simplify the Management of Access Rights

Deciding how best to use computer roles requires some upfront planning and configuration that might not be part of your initial deployment plan. To make effective use of computer roles, you also need to plan for and prepare appropriate role definitions for different sets of users. However, computer roles provide a powerful and flexible option for managing access to Server Suite-managed computers using your existing processes and procedures for managing Active Directory group membership.

For example, if you create a computer role group for Oracle servers and you deploy a new Oracle server, you simply add the computer account for the new server to the computer role group in Active Directory. If new database administrators join your organization, you simply add them to the Active Directory security group for Oracle database administrators. The computer role links the computer role group to the user role assignment and no additional updates are needed to accommodate organizational changes. If you need to modify the access rights, you can change the role definition and have the changes apply to all members of the group.

Because creating and managing computer roles is typically an ongoing administrative task after initial deployment, it is covered in the *Administrator's Guide for Linux and UNIX*. <!--TODO: xref ->

Migrating And Managing Service Accounts

After you have migrated accounts for the users who log on to the UNIX computers in your organization, the next step in the deployment is to decide how you want to manage the service accounts for applications in your environment. This section describes the options available and why you should consider migrating local service accounts to Active Directory.

Why Migrate Service Accounts?

A service account is typically a local user and group account dedicated to a specific application or to performing specific operations. In many cases, the service account has escalated permissions that allow it to run privileged operations on behalf of the application it supports. In addition, service accounts often have no password or a password that is wellknown to multiple users. Service accounts without a password typically require a local sudoers policy to control access. Service accounts with a shared passwords present a security risk because users can avoid an audit trail and, if passwords are managed locally, they may not conform to the password policies that are enforced for normal user accounts.

Therefore, the primary reason for migrating service accounts to Active Directory is to provide better security for accounts that can execute privileged commands, start and stop processes, or run specialized jobs on computers in your network.

Note that not all organizations choose to migrate and manage service accounts in Active Directory. There is no technical requirement that you do so. However, Server Suite provides you with several options for improving the security for service accounts. You should consider the options available, then decide which, if any, are most applicable for your environment.

Identifying Service Accounts to Migrate Tto Active Directory

Every UNIX platform has its own set of standard service accounts that are installed by default. For example, most UNIX platforms include services accounts for common applications, such as gopher, mail, ftp, and uucp. For most of these standard service accounts, there's no business reason to map them to accounts in Active Directory, unless you are trying to eliminate all local accounts on your UNIX computers.

In most cases, you can skip migration for the standard service accounts included by default when you installed the operating system as described in Eliminate default system accounts.

However, service accounts that run or manage applications or own an application's files are typically good candidates for mapping to Active Directory users. For example, an Oracle database instance has an oracle service account that owns the database server and the related processes that run in the background. Although usually linked to an application, a service account can also be account created to run scheduled jobs and own the files related to those jobs.

Service accounts that are good candidates for mapping to Active Directory users are ones that perform business operations without a password, rely on a shared password known to multiple users, or use shared SSH keys.

Service Accounts Without a Password

Most UNIX service accounts do not use passwords because UNIX services don't require an interactive log on to own files or run jobs. The most common way for users to access the service account is through the configuration of the sudoers file. The sudoers file provides rules that allow a subset of users to run the su command and change to the service account user. Mapping this type of service account to an Active Directory user eliminates the need for managing access through local sudoers policies and enables you to enforce the same password complexity rules for service accounts as normal user accounts.

Service Accounts with a Shared Password

The second most common way for users to access service accounts is with a shared account password. In this scenario, multiple users know the password for the service account and may be able to log on directly as that account. With shared accounts, there is no authoritative way to identify who is logging in to use the account. If you have any service accounts that rely on a shared password, you should consider migrating those accounts to Active Directory to eliminate the shared password.

Service Accounts that Use SSH Keys

Another common attribute of service accounts is that they often have a set of SSH keys that are available on multiple computers. The SSH keys allow the service account to transfer information from one UNIX computer to another without a password. In this scenario, a specific or the default SSH key for the service account exists in the authorized keys file on each of the computers to which the service account must connect.

Mapping a Service Account to an Active Directory User

After you identify one or more service accounts as candidates for mapping to an Active Directory user principal, you should identify an appropriate Active Directory user principal for the service account to map to. In most cases, there won't be an appropriate user already defined in Active Directory, so you will need to create one or more new users.

Create a New Active Directory User Account

You can use Active Directory Users and Computers or another tool to create a new user principal for each service account you are migrating to Active Directory.

> In most cases, you submit a request for a new account to be created using the procedure defined for your organization. For example, you might submit a request by filling out a service desk ticket and have the request serviced by a member of the account fulfillment team. The steps in this section only apply if you have permission to create new Active Directory user accounts. If you are not responsible for creating new Active Directory user principals, you can skip the following procedure.

To create a new Active Directory user for a service account:

1. Start Active Directory Users and Computers.
2. Expand the forest domain and the top-level UNIX organizational unit you created in Selecting a location for the top-level OU.
3. Select Service Accounts, right-click, then select **New > User**.
4. Type a name and account login information for the service account, then click **Next**.
5. Type and confirm the password to use for the service account in Active Directory, select the **User cannot change password** and **Password never expires** options, then click **Next**.

   The password must conform to your existing password policies for Active Directory users.

   If you are creating a new user to replace a shared account, type the password currently in use if it is acceptable within your site's Active Directory rules for password complexity. If you use the shared password, you should change the password after migration. If the current password is not complex enough, you should type a new password that complies or contact the Active Directory Enterprise Administrator for alternatives.
6. Click **Finish** to complete the creation of the new user principal.

Map the Unix Service Account to the Active Directory User

After you create a new Active Directory user principal for the service account, the next step is to map the UNIX account to the Active Directory user.

In preparation for this step, you should notify the user community that the service account will be unavailable for a brief period of time, so that you can make the change and verify that everything works as expected. You should then stop the service and any jobs associated with the service account.

By notifying users and making the service account unavailable for a period of time, you can prevent the change from affecting people who depend on the service to do their jobs. You can then use Access Manager to select the service account and map it to an Active Directory user.

To map the service account to an Active Directory user with Access Manager:

1. Start Access Manager.
2. Navigate to the UNIX user account
3. Navigate to the service account under a specific computer's Users node or under the Local Account Users node.
4. Select the service account, right-click, then click **Map to AD User**.
5. Type all or part of the Active Directory user name, click **Find Now**, then select the account in the results and click **OK**.

   Clicking OK updates the configuration on the remote host. You could accomplish the same thing by manually editing the configuration file (centrifydc.conf) or with a group policy.
6. Verify the service starts and executes operations as expected by switching to the root user or the service account and attempting to start the service.
7. Check for messages in the log files that the service account writes to. The entries should be regular service startup messages. You should verify that there are no errors or authentication failure messages.

   After you verify that the service starts as expected and that any jobs it owns start successfully, you can notify users that the service is available or do additional testing. Depending on your organization and the service account you have mapped to Active Directory, developers, database owners, application owners, and others may want to do full regression testing or execute specific test cases.

How the Mapped User Changes Your Environment

The Active Directory user you create for a service account must be enabled for authentication to work. However, enabling this new user account does present some potential risks to your environment. For example, on UNIX computers, creating the new user may have added a password for a service account that did not previously have one. If the new password is known to more than one person, the account may be considered a shared account and result in an audit finding.

Also, because the new account is a valid Active Directory user principal, anyone with the password can potentially log on to any Windows computer in the forest. By giving the service account a valid password and enabling the account, you have granted access to the Windows network for an account that previously had no access to Windows computers.

If you disable the account, you prevent that account from accessing all Windows and UNIX computers, running jobs, or executing required tasks. If you leave the account enabled and the password is compromised, both Windows and UNIX computers are vulnerable to attack. Even if the password is not compromised, failed password attempts could trigger an account lockout policy, rendering the service unusable.

Mapping service accounts to Active Directory users is a simple technique for managing access and password complexity for service accounts. If you have strong passwords and carefully control access to the account and its password, you can mitigate the risks. This strategy is also best suited to service accounts that already use a password. However, if granting the service account access to the Windows network presents too great of a risk, you should consider alternatives.

Creating a Service Account Role in a Zone

As discussed in How the mapped user changes your environment, mapping a service account to a Windows user makes the account vulnerable to attack. If the attack results in a guessed password, the attacker would be able to

log on as the service account, and, potentially, impersonate the service account on multiple computers on the network using SSH keys. Because the mapped user is also a valid Windows account, a successful dictionary attack might also grant access to Windows computers on the network. If the attack did not result in a guessed password, the failed password attempts could lock out the service account, making it unusable.

For service accounts that do not have a password, this vulnerability to a password-guessing attack would be a new security risk that did not previously exist. Therefore, simple account mapping is typically not the best solution for service accounts that are secured using sudoers policies or SSH keys instead of an account password.

If simple account mapping is not the appropriate solution for the service accounts in your organization, you may want to consider creating one or more service account roles. Roles enable you to securely manage the privileges of UNIX service accounts through Active Directory.

Create an Active Directory User Account for the Service

Because roles are tightly integrated with Active Directory user and group definitions, linking a service account to a role requires that you have an Active Directory user object to work with. In most cases, you should use your existing procedures to request a new user account. If you are responsible for creating new Active Directory user principals, see Create a new Active Directory user account.

Define a New Role with System Rights

It is recommended that you create the role definition for a service account in the appropriate child zone. If you want to make the role definition available in all child zones, however, you can create it in the parent zone. The specific selections you make for the role depend on the requirements of the service account for which you are creating the role definition. The steps described here provide general guidelines. Other settings may apply for the role definition in your organization.

To create a new role for a service account:

1. Start Access Manager.
2. In the console tree, expand **Zones** and the top-level parent zone.
3. Select the specific zone for which you want to define a role, and expand **Authorization**.
4. Select **Role Definitions**, right-click, then click **Add Role**.
5. On the General tab, type a name and description for the new role, then click **OK**.
6. Click the **System Rights** tab and select the following options that allow the service account to access UNIX computers using SSH keys or Kerberos, then click **OK**:
   - Non-password (SSO) login is allowed
   - Account disabled in AD can be used
   - Login with non-restricted shell

     In most cases, you should select the Login with non-restricted shell option. This option enables the service account to execute all of its commands in a standard shell. To have the service account run in a restricted shell, you must be able to identify and define rights for all of the commands that the service runs. The service account must also be able to execute all of its commands within the restricted shell (dzsh) environment. For most organizations, this additional security requires significant research and testing before it can be

implemented. However, forcing a service account to run in a restricted shell reduces the likelihood that a compromised service account could be used to attack computers on the network.

7. Select the new role, right-click, then click **Add Right**.
8. Select the login-all right for the zone, then click **OK**.

   This predefined right grants access rights for all PAM applications. If you determine that a specific service account should only use a specific PAM application, such as SSH or FTP, you can define a right that only allows that application to be used, then select that right in the role definition to specify that the service account must use the selected PAM application for access.

Create a Unix Profile for the Service Account and Assign the Role

After you define the role for the service account, you can create a UNIX profile for the service account. In most cases, you should define the UNIX profile for service accounts using machine-level overrides, rather than defining them for zones. Defining profiles for service accounts using machine-level overrides has the following advantages:

- Profile attributes are not affected the Zone Provisioning Agent. Most service accounts require specific UID and GID values. By specifying these values using a machine-level override, you don't have to worry about them changing when the Zone Provisioning Agent runs.
- You can explicitly identify which computers the service account can run on. If you define the UNIX profile for a service account in a zone, all of the computers in the zone are available to the same service account. If you define the UNIX profile using machinelevel overrides, the service account only runs on computers where it has a profile and the profile attributes for the service account can be different on different computers in the same zone.
- You can restrict the scope of the role assignments on a computer-by-computer basis. By defining the UNIX profile using machine-level overrides, you can configure different service account owners for development, testing, and production computers in the same zone.

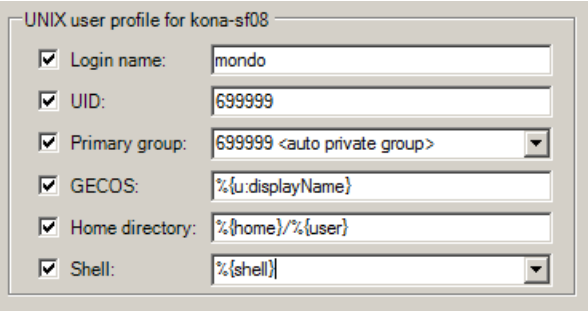
If the profile attributes are consistent across most of the computers in a zone and the service account should run able to run on all of those computers, you can define all or part of the UNIX profile for the parent or child zone to reduce the management of profile attributes on individual computers. However, if the legacy accounts had different attributes on different computers, it is typically best to use machine-level overrides.

To create the UNIX profile and assign the service account role using machine-level overrides:

1. Start Access Manager.
2. In the console tree, expand **Zones** and the top-level parent zone.
3. Expand the Child Zones node, select a specific child zone and expand it to display the Computers node.
4. Select computer for which you want to define machine-level overrides, right-click, then click **Add User**.
5. Click **Browse** to search for and select the Active Directory user account for the service, then click **Next**.
6. Select **Define user UNIX profile** and **Assign roles**, then click **Next**.
7. Select and define the attributes in the UNIX profile for the service account, then click **Next**.

   You can use inherited default values for any of the attributes from the default values specified for the zone or selectively override the default values for any of the attributes. For example if you define user defaults using

runtime variables in the zone, you can use the inherited values for the Login name, GECOS field, Home directory, and Shell and explicitly define the UID and primary GID for the service account profile.



8. Select the default UNIX Login role, click **Remove**, then click **Add**.
9. Click **Browse**, select the role you created for the service account, then click **OK**.
10. Click **OK** to accept the default start and end times for the role assignment, then click **Next**.
11. Review your selections, click **Next**, then click **Finish**.

Secure the Active Directory User Account

At this point, you have an enabled Active Directory user mapped to a UNIX profile for the service account. Having an enabled Active Directory user mapped to a service account, however, still presents a potential security risk as discussed in How the mapped user changes your environment. The next step is to decide how to secure the account to reduce the risk that it will be compromised and allow an attacker to gain access to the computers on your network. Depending on your organization's infrastructure and requirements, there are essentially two options available:

- Use SSH public key authentication
- Use Kerberos authentication

Each of these options has advantages and disadvantages and require different steps to configure.

Using Ssh Keys for Authentication

If you already use distributed SSH keys on hosts that run services, you can take advantage of that infrastructure and secure the service account by disabling the Active Directory user object in Active Directory. This is a common configuration that enables computertocomputer communication without a pass-phrase.

To use SSH public key authentication:

- Define the role for the service account with the **Account disabled in AD can be used by sudo, cron, etc** system right enabled. The role should not allow an interactive login.
- Ensure that the computers that communicate with each other have the SSH public key in the authorized_keys file.
- Select the Active Directory user principal you created for the service account and set the **Disable Account** option.

After you disable the account, it cannot be used for authentication on Windows or UNIX computers and it is not susceptible to a password-guessing attack. The account can continue to run UNIX services and use PAM-aware applications to communicate to other computers on the network using the SSH public key.

The primary advantage of using SSH authentication is that if you already have SSH public keys distributed to allow computer-to-computer communications, services should continue to work after the Active Directory user principal is disabled. There is very little configuration required to implement this solution.

There are, however, disadvantages to using SSH public keys. For example, you must manage key distribution. To allow a UNIX service account to communicate with other UNIX computers, you must generate the SSH key, and distribute that key to all of the hosts that the service account needs to communicate with. In addition, the most common configuration of SSH authentication allows keys to be used without a pass phrase. If the private key is compromised, an attacker could effectively impersonate the service account across multiple computers on the network and reacting to a compromised key can be timeconsuming because it requires you to remove the public key from every $HOME/.ssh/authorized_keys file distributed across the enterprise.

Using Kerberos Tickets for Authentication

If you are not already using distributed public-private SSH keys for authentication, you may want to consider using Kerberos authentication. Kerberos authentication is more secure than SSH keys and using Kerberos enables you to centrally manage access for service accounts, but it requires additional configuration.

To use Kerberos to secure UNIX service accounts:

- Install Kerberos-enabled software. You can use the Server Suite-provided version of OpenSSH or OpenSSH, version 3.9 or later, if it has been compiled with Kerberos support.
- Ensure the Active Directory user principal for the service account is enabled. Unlike SSH authentication, Kerberos requires the user account to be enabled to request ticket granting tickets or service tickets for other computers.
- Use the setspn.exe program to create at least one new Service Principal Name (SPN) for the UNIX service account.
- Run the adkeytab command on every UNIX computer where you want to re-use the Active Directory user principal that you created the new SPN for. This command creates a Kerberos keytab file that is only readable to the service account user. The keytab file allows the service account to request a ticket granting ticket so that it can communicate with other UNIX computers.
- Use the kinit command to request a ticket granting ticket from Active Directory for the UNIX service account. The ticket granting ticket (TGT) allows the service account to request additional host tickets for SSH communications to other UNIX computers.
- Use the klist command to list the tickets for the UNIX service account.

Testing And Migration

If you decide to use Kerberos for service accounts, you should test the computertocomputer communication. If you are migrating from authentication using SSH public-private keys, you can move ore rename the authorized_keys file for the service account on remote hosts to test authentication. After you test the Kerberos authentication of SSH communications and are certain that it works, you can delete the id_rsa, id_rsa.pub, and authorized_keys files for the service account that you have migrated.

Renewing Ticket Granting Tickets

Using Kerberos gives you consolidated control over UNIX service accounts. If an account is disabled in Active Directory, it cannot be used after any existing tickets expire.

If the service account runs scheduled jobs, you may want to create a crontab entry for the UNIX service account to run the kinit command at a regular interval so that the TGT doesn't expire. If the ticket expires and the kinit command isn't embedded in the job the service run, computer-to-computer communication will fail until the next time kinit is executed. For example, you may want to add logic to run klist to check whether there is a valid TGT, then run kinit is no valid TGT is found.

More Information

For information about using the setspn.exe program, see the Microsoft documentation for that program. For information about using Server Suite command line programs, see the corresponding man page.

Remove Local Service Accounts from Remote Computers

After you have migrated service accounts to roles in Active Directory, tested operations, and verified that commands and jobs run as expected, you can remove the local service accounts from remote computers to prevent them from being used.

For most organizations, removing local service accounts is a recommended security practice. However, you should leave the default operating system accounts as local accounts. In most cases, those accounts are not migrated to Active Directory. The default accounts for each platform are listed in the user.ignore file that is installed with the platform-specific Server Suite Agent.

Planning to Deploy in a Demilitarized Zone (DMZ)

Many organizations require both an internal network for corporate assets and a physical or logical subnet that exposes resources or services to a larger, external network, such as the Internet. For security, computers and resources in the external-facing perimeter network or demilitarized zone (DMZ) have limited access to the computers in the internal network. Because communication between the computers in the DMZ and the corporate network is restricted and protected through the use of a firewall, there are specific constraints on configuring authentication and authorization services.

This section describes how to deploy Server Suite components in specific DMZ scenarios.

Identifying the Computers to Protect

The computers that are most vulnerable to attack are computers that provide services such as e-mail, host external-facing web applications, and manage network routing through Domain Name Servers (DNS). Computers that provide these services are typically isolated from the internal network on their own subnet and allowed to communicate with the internal network through specifically designated channels. This configuration allows computers in the DMZ to provide services to both the internal and external network, but controls the traffic allowed to be routed between the computers in the DMZ and the internal network clients.

Creating a Forest and Trusts for a DMZ

It is recommended that you create a separate Active Directory forest for the computers to be placed in the network segment you are going to use as the demilitarized zone. You should then establish a **one-way outgoing trust** from

the internal forest to the DMZ forest.

Defining a one-way trust allows existing internal forest users to access resources in the DMZ without separate credentials or being prompted for authentication. The one-way trust also prevents any accounts defined in the DMZ forest from having access to the internal network. Accounts defined in the DMZ forest can only access computers inside the DMZ domain. If a privileged account in the DMZ forest is compromised, that compromise is limited to the scope of the DMZ forest.

For Server Suite, the one-way trust enables you to:

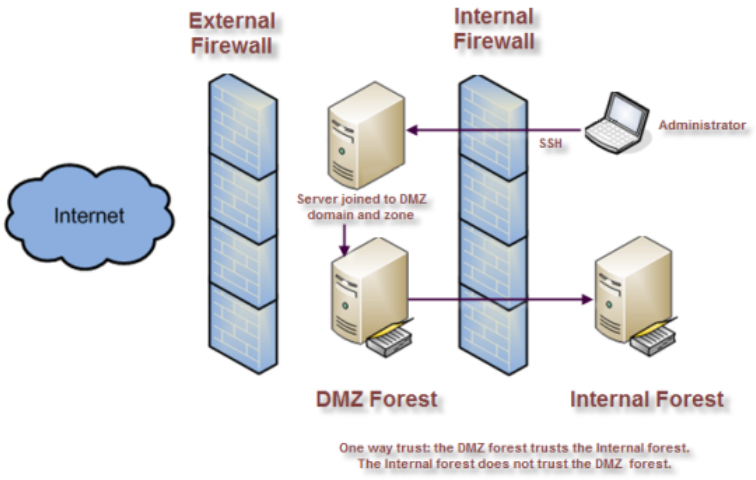
- Use the internal forest for authentication and authorization services for user accounts.
- Define computer accounts in the DMZ domain without permission to read data from the trusted domain of the internal forest.

In most cases, you should not use an existing Active Directory domain when deploying Server Suite Agents in a DMZ. Using an existing domain requires opening additional ports through the internal firewall to allow computers to connect directly to the domain controllers in the internal forest. Allowing computers in the DMZ to connect directly to the internal forest implicitly grants access to resources behind the internal firewall.

Defining Zones for Computers in the DMZ

You should create one or more zones in the DMZ forest and specify those DMZ zones when computers join the DMZ domain. You should also create and manage UNIX group profiles in the DMZ domain. However, you should define user accounts and UNIX profiles for in the internal forest and not the DMZ forest. Defining user accounts in the internal forest ensures that users can use a single password to authenticate across all network resources, including those in the DMZ.

The following figure provides an overview of the basic architecture for deploying Server Suite Agents when you have computers in a DMZ.



To enable authentication for resources in the DMZ, users must have:

- A valid Active Directory user principal.
- A complete UNIX profile in one or more zones in the DMZ.

- A UNIX Login, listed, or custom role assignment that allows access computers in the DMZ.

Creating a Firewall and Securing the Network

You should establish firewall rules that allow communication from the DMZ domain controllers to the internal domain controllers. Other computers in the DMZ, for example, the UNIX servers you want to isolate from the internal network, should have limited access to the corporate network. The most common exception is to allow communication from the corporate network to the DMZ network using port 22.

The client and server port requirements to enable communication through the firewall depend on the Windows operating system you have installed on the domain controllers and the functional level of the forest. For information about the specific ports to open and the services that use the ports, see Microsoft Active Directory documentation, such as *How to configure a firewall for domains and trusts*.

In addition to configuring a firewall that minimizes exposure to the corporate network, you should remove insecure network protocols, if possible, or replacing insecure authentication programs, such as POP3 and FTP, with Kerberos-enabled programs. These security steps reduce the potential for password exposure and the risk of an account in the corporate domain being compromised.

How to Join a Domain with a Read-Only Domain Controller (RODC)

With Windows Server 2008, you have the option of installing read-only domain controllers (RODC). Read-only domain controllers enable you to deploy a domain controller that hosts read-only partitions of the Active Directory database. Deploying a read-only domain controller enables you to make Active Directory data and reliable authentication services available in locations that cannot ensure physical security required for a writable domain controller.

You can also deploy read-only domain controllers to handle special administrative or application management requirements. For example, you may have line-of-business applications that are required to run on a domain controller, or application owners who must have access to the domain controller to configure and manage operations but not allowed to modify Active Directory objects as they could with a writable domain controller. You can grant a non-administrative domain user the right to log on to the read-only domain controller while minimizing the security risk to the Active Directory forest.

For more information about read-only domain controllers, see the *Read-Only Domain Controller (RODC) Planning and Deployment Guide*.

To join a domain that has a read-only domain controller:

1. Create a computer account for the computer in the DMZ that will connect to the readonly domain controller using a writable domain controller as described in Creating computer objects for the target set of computers.

   > You can create the computer account using the Access Manager console, an adedit script, or using the adjoin command with the --precreate command line option. However, be sure to create the computer account in a DMZ zone.

2. Use Active Directory Sites and Services or the repadmin program to replicate the computer account in the read-only domain controller. For example,

- In the console tree, expand Sites, and then expand the site of the domain controller that you want to receive configuration updates.
- Expand the Servers container to display the list of servers that are currently configured for that site.
- Double-click the server object that requires the configuration updates that you want to replicate.
- Right-click **NTDS Settings** below the server object, and then click **Replicate configuration to the selected DC**.
- In the Replicate Now message box, click **OK**.

3. (Optional) Open a Command Prompt and use the repadmin /showrepl command to verify successful replication on the read-only domain controller.
4. Block the route from the UNIX computer to the writable domain controller, if necessary.
5. Run the adjoin command with the self-service option. For example:

   adjoin mydomain.local --password c%ntrify --name quad90 --selfserve

   Because you have already created the computer account in Active Directory, you don't need to specify the zone to join the domain.

Enabling NTLM Authentication through a Firewall

Having a domain controller in the perimeter forest trust the internal domain requires you to open up ports through the firewall. The specific port requirements depend on the Windows operating system version and functional level of the forest. As an alternative, you can use NT LAN Manager (NTLM) authentication to allow Active Directory users in the internal forest to log on to computers in the perimeter forest.

Using NT LAN Manager (NTLM) authentication enables you to have a more restrictive firewall with a one-way forest trust between the perimeter forest and the internal forest. For example, if the firewall prevents you from using the ports required for Kerberos authentication or if you have limited communications between the forests to a specific port, you can use NTLM authentication to pass authentication requests from the domain controllers in the perimeter domain to the internal domain controllers through the transitive trust chain.

This configuration still requires a one-way trust relationship between the internal forest and domain controllers outside of the firewall.

Configuring the Domain Controllers that Allow NTLM Authentication

You can use the pam.ntlm.auth.domains configuration parameter to specify the domain controllers in the DMZ forest that should use NTLM authentication. This parameter requires that you specify the domain controllers using their Active Directory domain names. In addition to setting this parameter, you must be able to map NTLM domain names to their corresponding Active Directory domain names to support looking up user and group information in the cache.

Configuring a Map that Converts NTLM Domains to Active Directory

For Server Suite to automatically construct this map, it must be a able to send LDAP search requests to the domain controllers in the corporate forest. If the firewall restrictions will block these search requests, you must manually define a topology map that converts NTLM domain names into Active Directory domain names. To manually

configure how Active Directory domain names map to NTLM domain names, define entries in the /etc/centrifydc/domains.conf file using the following format:

ActiveDirectory_Domain_Name: NTLM_Domain_Name

For example:

arcade.com: ARCADE
ajax.org: AJAX

You can refresh the list of domain controllers in DMZ forest at any time by modifying the configuration parameters, then running the adreload command.

Managing and Evolving Operations After Deployment

In previous sections, you prepared for deployment, migrated existing users, configured automated provisioning for new users and groups, and added one or more custom roles for privilege management. Most of these activities are related to the initial deployment and extending deployment to additional sets of target computers and interactive users.

This section discusses management activity, evolving operational security, and adding services to extend authentication and authorization after deployment. Often, at this stage, the deployment project team begins to transition activity to an operations team or support staff.

Understanding How Server Suite Software Affects Operations

Through Active Directory, Server Suite software provides a consolidated solution to authentication, authorization, and policy management for Linux, UNIX, and Mac OS X computers. Because of this consolidation, however, you may need to make changes or additions to the IT tasks or operational procedures you currently have in place. Therefore, when you deploy Server Suite software in a production environment, you should consider how it impacts the management tasks typically performed by operations staff members.

The routine tasks that may be affected by adding Server Suite software to the environment fall into the following categories:

- Change management
- System administration
- Security administration
- Service desk operations
- Capacity management

Understanding Change Management Activities

Change management typically involves testing and installing updates to the operating system or installed applications. For example, most organizations follow a controlled process for reviewing and implementing changes to the operating system because of patches or new releases.

If you are preparing to update the operating system, the support staff should also plan to test that user log-ons and role assignments continue to function correctly after update. If a system patch or update affects the operation of Server Suite software, you should contact Customer Support to determine whether the patch is supported.

Staff members should also periodically review new and maintenance releases of Server Suite software to get the latest features, fixes to known issues, and enhancements requested by Server Suite customers. After downloading the software, you can review the release notes included in each package to determine what's changed and the suitability of the update for your organization.

Understanding System Administration Activities

Most system administration tasks involve managing users and groups. After deploying Server Suite software, this information is centralized in Active Directory for both Windows and non-Windows computers. Therefore, any administrative action for a user account affects that user on both Windows and UNIX computers. For example, if you disable a user account in Active Directory, the user will be unable to log on to any Windows or UNIXbased computer in the forest.

Typically, there is a period of time where staff members must use one set of steps for provisioning users and groups on the computers not yet joined to the domain and another set of steps for provisioning users and groups on computers that have successfully joined the domain. After you complete the migration to Active Directory, you can leverage the processes and tools you use for provisioning Windows users and groups for ongoing administration of UNIX users. For example, you can use Active Directory Users and Computers, in-house custom scripts, Access Manager, ADEdit, or another management tool to perform administrative tasks.

After deployment, you should prepare any site-specific or platform-specific instructions the operations staff should follow if you are making changes to the processes or tools they are familiar with.

Understanding Security Administration Activities

Security administration involves ensuring that operators are granted the appropriate rights for administering the computers and attributes that are required as part of their job but are prevented from accessing or changing computer settings outside their areas of responsibility.

Delegated Administration

Server Suite enables administrators to explicitly delegate management tasks to the appropriate users and groups on a zone-by-zone basis.

Password Policy Enforcement

Server Suite enables you to use existing Active Directory password policy rules, such as the minimum password length, complexity requirements, expiration, and allowed number of logon failures to allow before locking an account for both Windows and UNIX users.

Privileged Account Management

Server Suite enables you define rights, roles, and role assignments to control what users can do and who can execute privileged commands. You can also map privileged local accounts to Active Directory accounts to ensure better password security for those accounts. For example, the local root user account has full access to all data and can manipulate all settings on a UNIX computer. Mapping the local root user to an Active Directory user account enforces the Active Directory password policies on the account and makes it more difficult for an unauthorized user to obtain escalated privileges on the UNIX computer.

Understanding Service Desk Operations

Active Directory and Server Suite simplify help desk and service desk operations by:

- Enabling centralized administration for tasks such as adding new users or granting access to new computers.
- Consolidating user passwords and reducing the need for password resets.
- Simplifying troubleshooting for log on failures for UNIX users.

You should provide help desk staff with troubleshooting instructions to help them diagnose and resolve failed authentication or authorization tickets.

Understanding Capacity Management Activities

During the deployment of Server Suite Agents, you should monitor and analyze network traffic and domain controller replication to determine how well your environment handles the extra load of UNIX users and computers in Active Directory. In general, Server Suite software is configured to use minimal system resources and network bandwidth. In practice, however, you should monitor and evaluate the volume of traffic to determine its impact on performance across the network and the performance experienced by users logging on to UNIX workstations and servers.

If the network traffic or resource usage exceeds your expectations, you may want to modify the default configuration to better suit your network topology. For example, Server Suite provides numerous group policies and configuration parameters that you can modify to optimize network activity or control how much data is stored locally on individual computers.

Determining Whether You Need More Resources

In most cases, deploying Server Suite software does not noticeably affect the performance of the network or domain controllers. However, if you have a widely distributed network or replication delays, you should analyze your network's capacity to handle the additional load of UNIX users and computers to determine whether you need to make changes to ensure optimal performance and availability. For example, the following factors may require you to allocate additional resources:

- If the UNIX computers are in a different physical location than the domain controllers that they access, you may want to install a domain controller on a computer that is physically closer to the UNIX computers to reduce long-distance network traffic and the chance of replication delays.
- If you need to ensure availability in the event of a network or server failure, you should ensure that you have an adequate number of domain controllers to support the UNIX computers when they need to fail-over to a backup domain controller.
- If you add a large number of UNIX users to the Active Directory domain, apply your standard method for balancing domain controllers per number of users.
- If you add a large number of UNIX computers to the Active Directory domain, apply your standard method for balancing domain controllers per numbers of computers.
- If you move a large number of UNIX users and groups from a local directory (/etc/passwd and /etc/group) to Active Directory, you may need additional network bandwidth because authentication and authorization requests are now done over the network.

For more information about modifying configuration parameters, see the *Configuration and Tuning Reference Guide*.

Understanding How Caching Facilitates Lookups

Server Suite Agents store credentials in a local cache to reduce the network traffic required to look up information in the directory. For example, if a user executes the directory listing command in a UNIX command shell (such as with the ls -l command), the command looks up and displays a listing of files along with their attributes, such as the owner of each file.

However, a file's owner is stored as a number—the user's UID—on UNIX-based computers, but because the ls command displays the owner as a name and not a number, the ls command must look up the actual user name associated with the file owner's UID. Because UNIX UIDs and user names are stored in Active Directory, this lookup request must be serviced by Active Directory. If a large number of files are displayed when the ls command is run, this creates a substantial amount of lookup traffic between the UNIX computer and the Active Directory domain controller.

Server Suite reduces this traffic by caching the lookups so that the information does not have to be retrieved from the Active Directory each time a lookup is required. Commands such as ls check the local cache first for the relevant information instead of retrieving the information from Active Directory every time.

Troubleshooting Logon Failures

If a user attempts to log on to a computer that is in a Server Suite zone and the logon fails, the problem is typically caused by one of the following:

- Users attempting to log on to a computer they are not authorized to use.
- Users have an incomplete profile in the zone where the computer they are attempting to use is located.
- Users have not been assigned an appropriate role that allows logon access.
- Users have typed their non-Active Directory password or typed the wrong password more times than allowed.
- Local or group policy settings are applied to the computer to prevent access.

To investigate these potential problem areas:

1. Check whether the local UNIX computer can connect to the Active Directory domain controller.
   - Log on to the computer using a locally authenticated user, such as the local root user.
   - Run the ping command with the name of an appropriate domain controller in the forest.

     For example, if the local computer is joined to the snowline.org forest, the command might look similar to this:

     ```
     su -
     Password:
     ping shasta.snowline.org
     ```

     If the command receives a reply from the domain controller, the DNS service is functioning and the local computer is able to locate the domain controller on the network. If the ping command does not generate a

reply, you should check your DNS configuration and check whether the local computer or the domain controller is disconnected from the network.

2. Check Active Directory information by running the adinfo command. The output from this command should appear similar to the following:

   Local host name: magnolia
   Joined to domain: snowline.org
   Joined as: magnolia.snowline.org
   Current DC: shasta.snowline.org
   Preferred site: Default-First-Site-Name
   Zone: snowline.org/Acme/Zones/cascade
   Last password set: 2017-12-21 11:37:22 PST
   CentrifyDC mode: connected

   If the mode is disconnected, check whether adclient is running and network connectivity. On a slow network adclient may drop the connection to Active Directory if there is a long delay in response time.

   If the output displays an <unavailable> error, you should try running the adleave command to leave Active Directory, re-run the adjoin command, then re-run the adinfo command. For example:

   adleave --force
   adjoin --user skip --zone cascade snowline.org
   Password:
   adinfo

   If a problem still exists, check the DNS host name of the local computer and the domain controller, the user name joining the domain, and the domain name you are using.

3. Check the clock synchronization between the local UNIX computer and the Active Directory domain controller.

   If the clocks are not synchronized, reset the system clock on the UNIX computer using the date command.

4. Check for denied users and groups in the /etc/centrifydc/centrifydc.conf file or the Login Controls group policy. For example, open the centrifydc.conf file in a text editor, such as vi:

   vi /etc/centrifydc/centrifydc.conf

   - Search for the pam.deny.users line and make sure that the user who is trying to log on is not listed.
   - Search for the pam.deny.groups line and make sure that the user who is trying to log on is not a member of any group that is listed on this line.

5. Check the contents of the system log files or the centrifydc.log file after the user attempts to log on. You can use information in this file to help determine whether the issue is with the configuration of the software or with the user's account.

6. Check for conflicts between local user accounts and the user profiles in Active Directory by running the getent command. For example:

   getent passwd

   This command displays a list of local and Active Directory users with access to the computer. If the user's name is not listed but other Active Directory users are listed, the problem may be in the user's Active Directory account settings or UNIX profile.

If no Active Directory users are listed in the output of the command, check whether adclient is running and whether the Active Directory domain controller is available.

7. Check the user's Active Directory account settings using Access Manager or Active Directory Users and Computers. For example:
   - Check whether the user has a UNIX profile for the local computer's zone.
   - If the user has a UNIX profile in the zone, check whether the profile is currently enabled.
   - If the user has an enabled UNIX profile, check the user's group membership to determine whether it is a local group defined in a different domain than the computer account.
   - Check whether the user's account has been disabled or locked.
   - Check whether any user-based group policies have been applied to the user account that may prevent access to the UNIX computer.
8. Enable logging of adclient activity using the addebug command. For example:

   /usr/share/centrifydc/bin/addebug on

   This command enables extensive logging of each operation performed by the adclient process in the /var/log/centrifydc.log file. You can use the information in this file to further diagnose the cause of any problems or to enable Server Suite's support staff to assist with resolving any issues.

Evaluating Additional Services And Integrations

After you have deployed Server Suite Agents to implement an Active Directory-based security and directory services, you may want to explore other ways to take advantage of Active Directory's infrastructure. In evaluating ways to extend your security and directory services, you must first understand:

- How the UNIX servers and workstation that are joining the domain are used
- Which applications are accessed locally and which applications are accessed by remote users
- How the servers and workstations are managed, and whether administrators are local users or remote users
- Whether there are specific additional IT services you want to enable
- Whether there are specific controls you want to apply

As a starting point, you should consider whether computers joining the domain are workstations that primarily support local logon sessions or servers that require few, if any, local logon sessions. In many cases, UNIX computers have few interactive users but are frequently used as application servers that host database or web applications. For those computers, you should determine whether Active Directory authentication and authorization is primarily for administrators who manage the server or for users who log in to access the application.

Some of the ways you can extend and evolve the deployment of Server Suite software include:

- Adding authentication service for applications
- Adding custom reports for auditing UNIX properties
- Adding group policies
- Adding support for NIS clients

- Using programs optimized for Kerberos authentication
- Integrating with products from other vendors

Adding Authentication Service for Applications

Because Active Directory and Server Suite use Kerberos and LDAP standards, many Kerberos-enabled or PAM-enabled applications can use Active Directory for authentication and authorization service with little or no configuration. One way you can evolve your deployment is to add support for single sign-on to additional applications.

Supporting Single Sign-on for Kerberos-enabled Applications

The primary way that Server Suite supports single sign-on is through Kerberos. Kerberos provides a ticket-based authentication mechanism that is the default method for authentication in Active Directory. When a user logs on to a computer that uses Active Directory authentication, a Kerberos ticket is issued to the user and that ticket allows the user to access data, applications, other computers, and other sessions without having to present credentials again. This silent authentication that takes place in the background as users browse network shares or run applications is the key to enabling a single sign-on experience.

Many applications are Kerberos-enabled by default or can be configured to support the use of Kerberos tickets. By default, when a computer joins an Active Directory domain, Kerberos requests are forwarded and serviced by the Kerberos Key Distribution Server on the Active Directory domain controller. Therefore, in most cases, existing Kerberos-enabled applications can authenticate and authorize access without any modification.

If you use an application that is not configured to use Kerberos authentication by default, however, you may need to modify configuration options or use specific command line options to support single sign-on.

In addition, users must be assigned to a role with the **Non-password (SSO) login is allowed** system right. This right is enabled in the predefined UNIX Login role. If you create custom roles and want to allow single sign-on, you should select this system right when defining the role.

Supporting Single Sign-on for PAM-aware Applications

Pluggable Authentication Modules (PAM) provide a flexible mechanism for authenticating users regardless of the underlying authentication system. Most programs and applications that rely on user authentication use PAM.

The agent uses its own PAM module, pam_centrifydc.so, to direct PAM requests to Active Directory. Therefore, in most cases, existing PAM-enabled applications can authenticate and authorize access without any modification and support single sign-on without any special configuration.

One known exception, however, is that most database applications support PAM authentication, but do not enable it by default. To support single sign-on for database applications, you should modify the database configuration to enable PAM authentication.

In addition, users must be assigned to a role with the **Non-password (SSO) login is allowed** system right. This right is enabled in the predefined UNIX Login role. If you create custom roles and want to allow single sign-on, you should select this system right when defining the role.

Supporting Active Directory Authentication for Apache and Java Applications

Most Web and J2EE platforms provide their own native authentication and authorization services for Web developers to use. With Server Suite, you can choose to extend the native interfaces to enable web applications to

provide single sign-on capability or redirect authentication requests to Active Directory instead of a native authenticator.

Supporting Database Server Applications

Most database servers provide their own native authentication and authorization services. With Server Suite, you can choose to extend the native interfaces to enable supported database servers to provide single sign-on capability or redirect authentication requests to Active Directory instead of a native authentication service.

Adding Custom Reports for Auditing Unix Properties

Server Suite includes several default reports that you can use to monitor and audit access to the computers in your environment. The default reports provide detailed information about your UNIX users, groups, computers, zones, and licenses, and enable you to verify which users have access to specific computers, zones, or applications. Default reports also provide easy access to the information that you require for auditing, business planning, and regulatory compliance. After you generate a report, you can save the report in the following formats:

- Microsoft Excel (.xls)
- Microsoft Word (.doc)
- Adobe Acrobat (.pdf)
- XML document (.xml)

For example, after generating a report with information about the users in each zone, you can save it as a Microsoft Excel spreadsheet (.xls), and import the information into an Excel Workbook to create a Charge Back report on account usage for each department.

One of the most common ways to evolve the Server Suite deployment is to create custom reports that are specifically tailored to your organization and auditing requirements. The Access Manager console includes a Report Wizard that allows you select the specific Active Directory objects and properties and the relationships on which you want to base the report.

For information about creating and generating custom reports, see the *Administrator's Guide for Linux and UNIX*.

Adding Group Policies

For many companies, centralized policy management and configuration control is just as important as centralized identity management. With Server Suite, you can apply group policy settings from Active Directory to the non-Windows computers and UNIX users.

Evaluating Existing Policy Settings

If you have applied any domain-wide policies in the Active Directory forest, you should review what the policy settings are and where they are enforced for Windows-based computers. You should then evaluate which policy settings, if any, are applicable for computers running UNIX, Linux, or Mac OS X operating systems. For example, most organizations establish a policy for password complexity. You can view your current password policy settings by clicking **Domain Security Policy** under **Administrative Tools** to open the **Default Domain Security Settings**, then select **Password Policy**.

If you enable any password policy settings for the domain, they automatically apply to UNIX users and managed computers because Active Directory uses these settings when authenticating users. If you enable or change any of

the default domain policy settings, you should consider how they affect UNIX users and computers. For information about the standard Windows group policies that apply for UNIX, see the Group Policy Guide.

Adding Server Suite-specific Group Policies to a GPO

You can add Server Suite-specific configuration settings to any Group Policy Object applied to any site, domain, or organizational unit in the Active Directory forest. You can then manage the specific policies enabled and settings applied centrally through the Group Policy Object Editor on Windows.

Each GPO can consist of configuration information that applies to computers, configuration information that applies to users, or sections of policy that apply specifically either to users or to computers. You link a GPO to an Active Directory organizational unit, domain, or site. Windows then applies the policy settings based on an established hierarchical order.

The Server Suite-specific group policy settings available for users and groups are defined separate administrative templates (.adm or .xml files) that can be added to any GPO. If you enable any of the policy settings, they are written to a virtual registry on the UNIX computer. The Server Suite Agent then runs a set of local mapping programs that read the virtual registry and modify local configuration files to implement the setting defined by the group policy. You can also create your own custom administrative template and mapper programs to implement custom group policies.

For more detailed information about creating and managing Server Suite-specific group policies, see the *Group Policy Guide* and Active Directory documentation.

Adding Support for NIS Clients

You can extend Server Suite software to provide NIS service from a Server Suite-managed computer, acting as a NIS server, to NIS client requests using Active Directory as the central data store for NIS maps.

There are many scenarios in which adding the Server Suite Network Information Service (adnisd) to your infrastructure can enable you to integrate Server Suite and Active Directory with other enterprise solutions. For example, the adnisd Network Information Service and Server Suite zones can be used to centrally manage and map multiple UNIX identities to a Windows user account for access resources stored on EMC Celerra Network Servers or Network Appliance Filers. Active Directory remains the central identity store and zones remain the primary way of mapping UNIX profiles to a user account, but the Server Suite Network Information Service enables you to deliver the appropriate information to servers and devices across the network.

Using the Server Suite Network Information Service in conjunction with the Server Suite Agent is a scenario like this provides the following advantages:

- **Redundancy.** As an NIS client, the Celerra Network Server can find NIS servers by broadcasting on the local subnet. If a subnet hosts more than one Server Suite-managed computer acting as a NIS server, the Celerra or NetApp server can fail over from one NIS server to another NIS server on that subnet, thus enabling multiple NIS paths to the same data held within Active Directory.
- **Multi-domain support.** The Server Suite NIS service can provide user mapping data to NIS clients who may have an account anywhere within an Active Directory forest, including remote or child domains. Through the Active Directory Global Catalog, Server Suite Agents can find user mapping information for users anywhere across the forest.

Extending your deployment with the Server Suite Network Information Service also enables you to centrally manage network information and publish custom information to NIS clients throughout the network without modifying the underlying Active Directory schema.

Using Programs Optimized for Kerberos Authentication

As a management platform, Server Suite provides its own versions of commonly-used open source programs. For example, the following packages have been optimized to work with Server Suite software and Active Directory:

- Standard MIT Kerberos utilities, such as kinit and kdestroy, are installed with the agent to support Kerberos keytab management for accounts in Active Directory.
- Kerberos-enabled client programs such as OpenSSH, support Kerberos authentication and single sign-on for secure connections between Server Suite-managed computers.

Integrating with Products from Other Vendors

Server Suite software also integrates with products from many other vendors, such as Splunk, IBM DB2, SAP Netweaver and Secure Network Communication (SNC), and Quest ActiveRoles Server. In addition to Active Directory, you can use Server Suite software to extend other Microsoft services such as Services for Network File System (NFS), Microsoft Identity Integration Server (MIIS), and Microsoft Active Directory Federation Services (ADFS).

For more information about add-on packages, integration with other systems, or configuring Server Suite software to work with other products, see the Resource Center on the Delinea website.

Getting Assistance from Support

It is recommended that you take the following steps if you need assistance with an issue or have questions about the operation of Server Suite components:

1. Check the Support Portal on the Delinea website to search the Knowledge Base to see if your problem is a known issue or something for which there is a recommended solution.
   - Open http://www.centrify.com/support/login.asp in a Web browser.
   - Log in using your customer account information and password.
   - Click **Find Answer** and type one or more key words to describe the issue, then click **Find** to view potential answers to your question. For example, to search for known issues, type known issues and click **Find** to see articles related to the known issues in different releases.

     If your issue is not covered in an existing Knowledge Base article or the Server Suite documentation set, you should open a case with Delinea Support.
2. Click **Log a Case** to open a new case using the Delinea Support Portal.

   Alternatively, you can contact Delinea Support by email or telephone, if you prefer. Worldwide contact information is available in the “How to open a case and collect information for Delinea Support” Knowledge Base article (KB-0301).
3. Provide as much information as possible about your case, including the operating environment where you encountered the issue, and the version of the Delinea product you are working with, then click **Submit** to open the case.

Before or after opening a support case, you may need to collect additional information about your environment. To help ensure your issue gets resolved quickly and efficiently, you should take the following steps to gather as much information as possible:

1. Verify the Server Suite Agent is running on the computer where you have encountered a problem. For example, run the following command:

   ps –ef | grep adclient

   If the adclient process is not running, check whether the watchdog process, cdcwatch, is running:

   ps –ef | grep cdcwatch

   The cdcwatch process is used to restart adclient if it stops unexpectedly.
2. Enable logging for the Server Suite Agent. For example:

   /usr/share/centrifydc/bin/addebug on
3. Create a log file for the Mac OS X Directory Service. For example:

   killall –USR1 DirectoryService
4. Run the adinfo command to generate a report that describes the domain and current environment. For example:

   adinfo --diag --output filename
5. Duplicate the steps that led to the problem you want to report. For example, if an Active Directory user can't log in to a Server Suite managed computer, attempt to log the user in and confirm that the attempt fails. Be sure to make note of key information such as the user name or group name being, so that Delinea Support can identify problem accounts more quickly.
6. Verify that log file /var/log/centrifydc.log or /var/adm/syslog/centrifydc.log exists and contains data.
7. Generate information about Active Directory domain connectivity and configuration files by running the following command:

   adinfo --support

   This command writes output to the file /var/centrify/tmp/adinfo_support.txt.
8. If there is a core dump during or related to the problem, save the core file and inform Delinea Support that it exists. Delinea Support may ask for the file to be uploaded for their review.

   If the core dump is caused by an agent process or command, such as adclient or adinfo, open the /etc/centrifydc/centrifydc.conf file and change the adclient.dumpcore parameter from never to always and restart the agent:

   /etc/init.d/centrifydc restart
9. If there is a cache-related issue, Delinea Support may want the contents of the /var/centrifydc directory. You should be able to create an archive of the directory, if needed.
10. If there is a DNS, LDAP, or other network issue, Delinea Support may require a network trace. You can use Ethereal to create the network trace from Windows or UNIX. You can also use Netmon on Windows computers.
11. Create an archive (for example, a .tar or .zip file) that contains all of the log files and diagnostic reports you have generated, and add the archive to your case or send it directly to Delinea Support.

12. Consult with Delinea Support to determine whether to turn off debug logging. If no more information is needed, run the following command:

    /usr/share/centrifydc/bin/addebug off

Templates and Sample Forms

This section provides templates and samples that you can customize and use in the deployment process. These templates represent documents that are commonly used, such as change control requests and email notifications of software changes. Your organization may require you to use organization-specific versions of these documents.

Simplified Environment Analysis and Zone Design Template

This template provides a framework for the information that the deployment team should collect, analyze, and document in evaluating the existing network infrastructure and how it will change after deployment. Depending on your environment and requirements, you might need to collect additional information, but this template describes the most common elements with examples that you can adapt to your organization.

1. Introduction

   Use this section to provide a brief overview of the deployment plan. For example, document the features you plan to deploy, any primary goals that might affect design decisions, and any dependencies or special considerations, such as activities that require change control approval or enhanced permissions.

2. Network architecture

   Use this section to capture details about your existing network configuration and Active Directory architecture. For example, you might want to record information about the Active Directory site, forest, and domain controllers, including trust relationships and domain and forest functional levels, if applicable.

   You might also include details about your DNS configuration, including whether you have more than one DNS namespace and any port requirements, firewall restrictions, and any network connectivity issues. For details about the default ports used, see Default ports for network traffic and communication.

3. Server Suite-managed computers

   Use this section to provide details about the existing UNIX, Linux, and Mac OS X computers on which you plan to deploy the Server Suite Agent.

4. Provisioning process

   Use this section to describe the process for provisioning computers, groups, and users.

5. Rights, roles, and role assignments

   Use this section to describe the rights, roles, role assignments, and configuration policies you require. For example, if you use the sudo program and the sudoers file, use this section to document how rights and roles defined in the sudoers file and whether the sudoers file is managed locally on each computer or in a central location.

6. Zone architecture

   Use this section to identify the Active Directory schema you are using and where Server Suite-related objects are located in the Active Directory forest.

7. Deployment preparation in Active Directory

Use this section to summarize the deployment of Server Suite components into the existing Active Directory forest and domain.

8. Windows installation

   Use this section to describe how zones will be created and configured.

9. UNIX deployment

   Use this section to describes the deployment of Server Suite Agents on UNIX computers.

10. Group Policies

    Use this section describes the group policies that will be deployed for UNIX computers.

Change Control Request Form

Most larger organizations require a formal change request to be submitted for any changes to Active Directory. The purpose of this template is to illustrate a request for creating new Active Directory organizational units, groups, and users. If the deployment team is not allowed to add UNIX groups and group members to Active Directory after the organizational structure if created, it is likely the project will experience delays.

Computer:

Change Requested:

Approved By:

Test Case Matrix Sample

To validate the pilot deployment, most organizations execute at least some formal testing of features and functionality. The purpose of this template is to suggest a basic set of test cases to execute that apply to most environments. These test activities apply to setting up your environment, installing the software, and performing common administrative tasks. You can skip any activities that don't apply to your organization.

Testing Matrix

Activity	Remarks	Date
Create the OU Structure with a script or manually	Active Directory setup activities	
Create the OU Permissions with a script or manually		
Create Security Groups with a script or manually		
Create Distribution Groups with a script or manually		
Create the Zones Container with the Setup Wizard, a script, or manually		
Create the Licenses Container with the Setup Wizard, a script, or manually		

Create the service account for the Zone Provisioning Agent		
Update the local or domain policy to allow the Zone Provisioning Agent service to Log on as a service right		
Deploy the agent on computers		
Create a zone with a script or Access Manager console	Access Manager console activity	
Delegate zone control with a script or using the Delegate Zone Control Wizard		
Pre-Create Computer account		
Import UNIX groups from group files or group NIS maps		
Resolve mapping issues		
Import UNIX users from passwd files or passwd NIS maps		
Assign interactive users to the UNIX Login role	Authorization activities	
Assign users who need profile but not access to the listed role		
Join computers to the domain using adjoin	You should prepare for migration and create one or more initial zones before you join the domain.	
Configure root-equivalent rights		
Configure root-equivalent replacement role		
Add an Active Directory group for the role		
Test role access		
Test role privileges control		
Identify current management process (manual or automated)	UID consolidation activities	

Document the new management process		
Define the business rule for assigning UIDs (for example, SID)		
Identify active users to preserve, migrate, and keep		
Run adfixid to change file ownership		
Identify current management process (manual or automated)	GID consolidation activities	
Document the new management process		
Define the business rule for assigning the primary GID values (for example, GID)		
Identify the Active Directory groups for primary GID assignments	Domain Users	
Validate Active Directory log on credentials	User login activities	
Validate successful access to UNIX, Linux, Mac OS X		
Validate successful application usage		
Validate password complexity policy		
Validate account lockout policy		
Validate role enforcement		
Validate single sign on		
Validate password reset		
Test period users validated	Clean up activities	
Test period groups validated		
Test period roles validated		
Run adrmlocal to remove local accounts		

Preliminary Software Delivery Notification Email Template

The purpose of this template is to notify users that they are scheduled to receive new software that will be delivered to their computers. This email notice should include a specific delivery date or a time frame estimate, if possible.

Although you can delete this information from the email message you send out in your organization, this notice is most effective if users know specifically when the change is scheduled to occur. You can also customize the specific requirements or objectives that Server Suite is helping your organization achieve.

Colleagues:

The *[Department Testing Server Suite]* has successfully completed testing of the Server Suite software and is ready to begin the deployment portion of the project. The target date for deployment is *[Scheduled time]*.

Deployment of this software will greatly enhance our ability to comply with multiple industry requirements to include *[List objectives, such as: PCI, Sarbanes-Oxley compliance, Internal/External Security Audit, specific organization initiatives]*. These requirements are in alignment with prioritized corporate business objectives.

The Server Suite software enables the streamlining of authentication, access controls and privileges, and auditing for all corporate IT systems. For the most part, deployment and streamlined authentication and authorizations services occurs “behind the scenes” with minimal, if any, user disruption. You should not notice any operational changes when the software is deployed to your computer.

Thank you for your cooperation,

[IT Department Signature]

Department-specific Announcement and Instructions Email Template

The purpose of this template is to notify users in a specific department that they are scheduled to transition to using Server Suite for authentication and authorization. This email notification indicates that you plan to join the computers in the department to an Active Directory domain during down time. Depending on your organization’s policies, this email may suggest users log on with their Active Directory credentials or explicitly state that they can continue to log on with their existing credentials.

Colleagues:

The *[Specific department you are deploying to, such as: Accounting Department]* is scheduled to begin the transition to Server Suite next week. In order to ensure a smooth transaction we simply ask that you log off of all systems before leaving for the weekend. When you return to work the following week, you should *[be able to log on with your current user name and password]*.

If you experience any difficulties logging on, or with application connectivity, please submit a ticket or contact the support desk immediately. Several members of each department helped the IT team perform successful testing and validation of this new solution, and we anticipate a smooth transition.

Thank you for your cooperation,

[IT Department Signature]

General Announcement and Deployment Schedule Email Template

The purpose of this template is to notify a broader user community of the deployment schedule for multiple departments across the company. This sample also illustrates the type of notes you can incorporate into the email message to keep other groups informed of their status. The general announcement may also include portions of the other two email templates. For example, you may want to include the objectives the transition to Server Suite helps the company achieve or the instructions to use current or Active Directory credentials after migration.

Colleagues:

At the completion of the week, the *[Server Suite Deployment Project Team]* will allocate first response resources to the next department scheduled for deployment.

This is the schedule coordinated with the Department Heads throughout the company:

Date	DEPARTMENT	REMARKS
9 May 2017	Information Technology	
16 May 2017	Accounting	
23 May 2017	Marketing	Pending EOQ Reports
30 May 2017	Security	
6 Jun 2017	Sales	
13 Jun 2017	Executive	
20 Jun 2017	PMO	
27 Jun 2017	Data Warehouse	Pending EOQ Reports
3 Jul 2017	Training	
10 Jul 2017	Business Development	
17 Jul 2017	Audit	

The IT Department would like to thank everyone to date for their work on this project, and look forward to a successful deployment. If you have any questions, please submit them to the *[Server Suite_project]* distribution list and include your contact information. We will respond with answers or contact you directly for more information.

Sincerely,

[IT Department | Server Suite Deployment Project Team]

Deployment Team Task Checklist

Before you install the pilot deployment, you should prepare a deployment checklist to ensure you have the information you need to successfully complete the deployment. For example, you should review port requirements, verify DNS resolution, and create one or more spreadsheets that describe the user and group accounts to be imported and any special relationships, such as membership in specific groups that need to be preserved or any special configuration you want to implement.

Creating a deployment checklist is optional, but can help you to collect detailed information about each of the computers targeted for deployment.

The following example illustrates information you can collect and record in a deployment team task checklist.

Preparing computers for deployment	
	Operating system, version, and patch level for target computers
	Host name and IP address for target computers
	Current disk space for target computers
	Review the details of the current DNS configuration For example: Is the address resolved through a UNIX DNS server, Windows DNS server, or settings in the /etc/hosts and /etc/resolv.conf files? Is the computer using a DNS server that has SRV records for Active Directory domain controllers? Are UNIX subnets registered and associated with Sites in Active Directory? Are you using a disjointed DNS namespace, where a UNIX computer name may be server.company.com but the Active Directory domain name is server.windows.company.com? Are you using DNS aliases and do they resolve correctly? Are there multiple network interfaces (NIC) in use?
	Current network time provider (NTP) For example, does the computer use a different server to determine the time than the Active Directory domain controller?
	Current firewall configuration For example, are there any firewalls blocking required ports between the UNIX computer and the Active Directory domain controllers for the registered sites?
	Current applications and services For example, do you have Perl, Samba, or OpenSSH deployed? Are the versions you have compatible with the Server Suite Agent or—if a Server Suite version is available—to be replaced by versions provided by Server Suite? Do you have existing authentication providers deployed? Are existing applications and services Kerberos-enabled or PAM-enabled? Are there other applications that require local users or groups?
	Current source of user and group information For example, are the /etc/passwd and /etc/group files the only source of user information for the users who access this computer or other identity stores, such as existing LDAP servers or NIS domains, used? Are there any specific users or groups that should remain locally defined?
	Current NSS configuration For example, have you reviewed the contents of the nsswitch.conf file to check for other sources of user and group information?
	Connectivity between this computer and the domain controller For example, is there a reply from the domain controller when you run the ping command?
	User names and UIDs checked for conflicts across the target group
	Zone requirements analyzed for the target group

	Zone identified for this computer
	Server Suite Agent installed and the computer joined to the domain
	Groups allowed or denied access identified for this computer
	Existing users and groups for this computer imported into Active Directory
	Imported user and group profiles mapped to Active Directory accounts
	Allowed or denied groups configured using parameter values or group policy

If you use a deployment checklist, you can also include additional notes and details about the activities performed. For example, a partially completed checklist might look something like this:

Preparing computers for deployment	
	Operating system: Sun Solaris 10 with all patches applied (17-April-2017)
	Host name and IP address: aspen, 177.29.10.10
	Current DNS configuration: Resolved through the enterprise DNS server, spider.ajax.org
	Current time source is NTP server: ntpd on solstice.ajax.org Change for deployment: Use SNTP on the Active Directory domain controller
	Current firewall configuration: No port issues
	Existing OpenSSH version to be replaced, no other issues found.
	Current source of user and group information: /etc/passwd, /etc/group, and NIS domain nwest03 have users who access aspen
	Connectivity with the domain controller: Verified by JR (2-May-2011).
	User names and UIDs checked for conflicts across the target group: Analyzed by JR and DC (4-May-2017).
	Zone requirements analyzed for the target group: Zones required for the target group are nwest01, swest02, corp-main, and nwest03 (9 May 2017). SF to recommend new extended zone descriptions for approval.
	Zone identified for this computer: nwest03

	Server Suite Agent installed and the computer joined to the Active Directory domain: dc3colorado.ajax.org, OU: US-UNIX-Computers
	Groups allowed r denied access identified for this computer: Allowed access group—all_ employees, oracle_sys Denied access—consultants, temps
	Existing users and groups for this computer imported into Active Directory: Completed by DC (20-May-2017).
	Imported user and group profiles mapped to Active Directory accounts: Work complete for users and groups that already had matching Active Directory candidates. Work in progress for the remaining profiles without any matching Active Directory candidate. Target date for completion: 31-May-2017
	Allowed or denied groups configured using parameter values or group policy: TBD

Permissions Required for Administrative Tasks

This chapter describes the permissions required to perform administrative tasks that affect Centrify-specific objects in Active Directory. You can set these permissions manually for individual users and groups who manage Centrify zones in Active Directory. However, setting permissions manually can be time-consuming and error-prone. In most cases, you should use the Zone Delegation Wizard to authorize users to perform specific tasks.

At a minimum, all Access Manager actions require users to have generic Read permission. This permission is typically granted to all Authenticated Users by default.

Because Authenticated Users have read access, they can run reports in the Report Center. No additional rights need to be granted to enable users to run reports.

How Permissions Are Set

Access Manager requires specific rights for administrators to work with objects such as UNIX users, groups, and computers within Active Directory. As part of your deployment planning process, you should review the rights required to set up and manage Centrifyspecific objects and be familiar with how to manually assign rights for managing Centrify objects, if needed.

Built-in Windows groups, such as Domain Admins and Domain Users, have default permissions, which might be customized for your organization. In general, the administrators for the forest root domain have broad authority to set permissions for all other users and groups, including the administrators of other domains. Therefore, whether you can modify the permissions for specific users and groups within your Active Directory environment will depend on the policies of your organization.

If you have the appropriate authority, there are several ways you can access, verify, and modify the permissions assigned to specific users and groups.

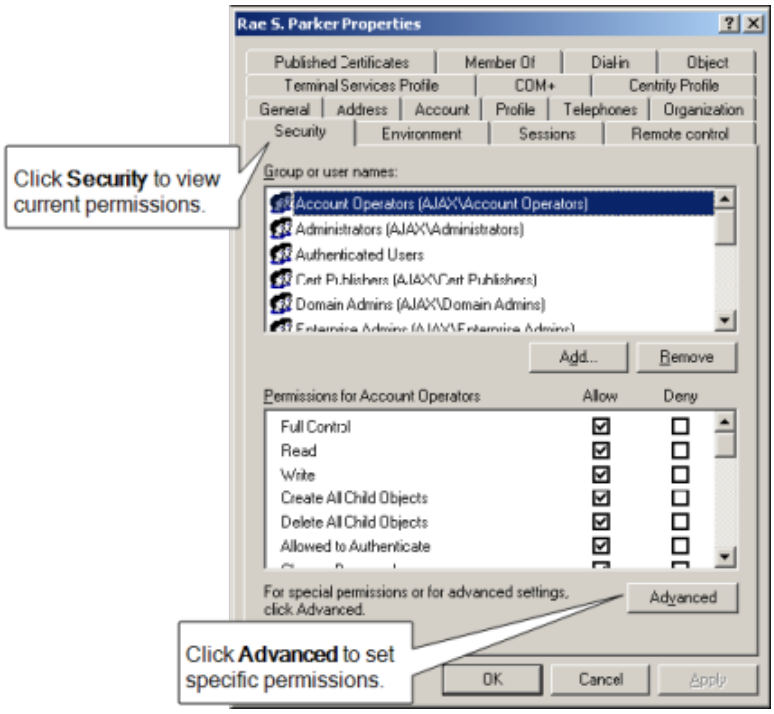
For example, you can view and modify permissions in the following ways:

- Use **ADSI Edit** to directly modify any Active Directory attributes.
- Use **Active Directory Users and Computers** to set basic or advanced permissions on any Active Directory object through the **Security** tab.

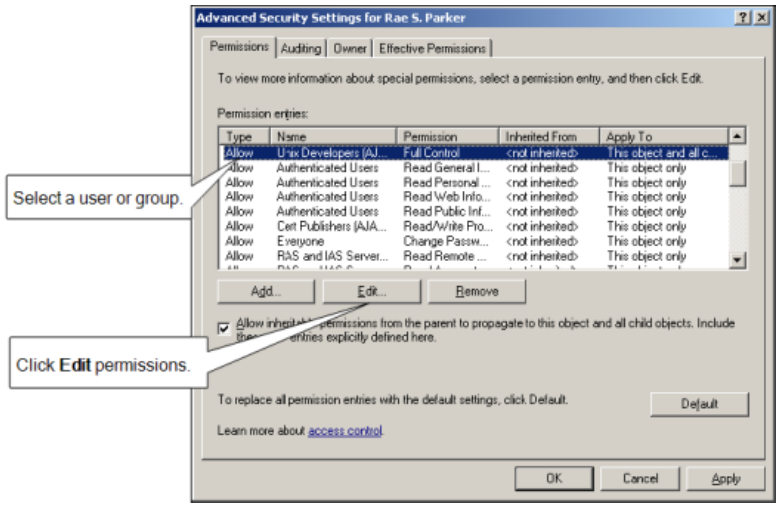
  To display the Security tab, select **View > Advanced Features**. To access some permissions, however, your user account must have **Create all child objects** or **Write all properties** permissions.
- Run the **Zone Delegation Wizard** to set the appropriate permissions for specific users or groups to perform specific tasks within a zone.
- Click **Permissions** when viewing Zone Properties in Access Manager to set basic or advanced permissions on any zone object.
- Click **Permissions** when viewing the Centrify Profile for a user in Access Manager to set basic or advanced permissions on any user object.

The following steps illustrate how you can set permissions from Active Directory Users and Computers:

1. Open the console and connect to the Active Directory domain.
2. Select an Active Directory object, such as a user or computer, rightclick, then click **Properties**.
3. Click the **Security** tab, then click **Advanced**.



4. Select the user or group to which you want to assign rights, then click **Edit**.

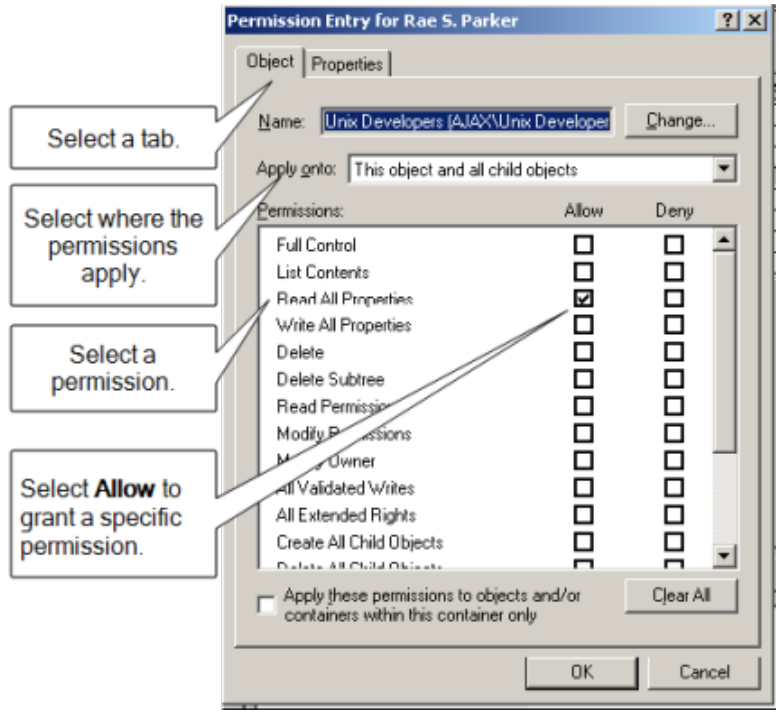


If the user or group to which you want to assign permissions isn't listed, click **Add** to find the account.

5. In the Permission Entry dialog box, click the **Object** or **Properties** tab, as needed.

   Selecting Object or Properties and where the permission should be applied varies depending on the task you are allowing a user or group to perform.

6. Select the specific rights you want to assign by scrolling to find the permission, then clicking the **Allow** check box.



7. When you are finished setting the appropriate permissions, click **OK**.

For more specific information about how to set permissions on Active Directory objects and properties and how to view, modify, and remove permissions, see your Active Directory documentation.

Permissions Required to Use the Setup Wizard

In most cases, you run the Setup Wizard to guide you through the configuration of Active Directory for Centrify. The Setup Wizard updates Active Directory with Centrify-specific objects and properties, including zone and license containers that are required for proper operation.

To successfully perform initialization tasks, the user account that runs the Setup Wizard must have specific rights. Because some of these rights might be reserved for administrative accounts, some users might be prevented from performing certain steps in the Setup Wizard.

To allow other user accounts to run the Setup Wizard, you can manually create the appropriate container objects, then assign to those objects only the specific permissions needed to correctly complete the configuration of Centrify-specific objects. Users can then use the Setup Wizard to select the appropriate container objects and perform all of the necessary steps without being members of an administrative group.

Licenses Container Permission Requirements

The following table describes the minimum rights that must be applied to the Centrifyspecific container objects or other users to successfully complete the configuration of Centrify software.

This target object	Requires these permissions	Applied to
Licenses container	Read all properties Create classStore Objects Modify permissions	This object only
	Write Description property Write displayName property	This object and all child objects
	The Setup Wizard requires you to create or select at least one parent container for license keys. By default, this container object is: domain/Program Data/Centrify/Licenses You can create additional License containers, if needed, through the Manage Licenses dialog box. By default, all Authenticated Users have read and list contents permission for the Licenses container and all of its child objects. You can change these permissions if you want to restrict access to Access Manager.	
Zones container or any container used as a destination for a new zone	Read all properties Create classStore Objects Create container objects	This object only
	Write displayName property	This object and all child objects

	The Setup Wizard requires you to create or select a parent container object for creating new zones. By default, this container object is: domain/Program Data/Centrify/Zones You can use other containers for zones, if needed. For example, if you have created a separate high-level organizational unit called UNIX as the parent container: domain/UNIX/Zones	
ZoneName/Computers container	Create group objects Write Description property	This object only
	These permissions are only needed if you are supporting “agentless” authentication in a zone.	
Computers container For example, the generic Computers container: domain.com/Computers	Write operatingSystem property Write operatingSystemVersion property Write operatingSystemHotfix property Write operatingSystemServicePack property	SELF on Computer objects
	These permission are granted to each computer’s SELF account when you select the Grant computer accounts in the Computers container permission to update their own account information option in the Setup Wizard.	

Licenses Container Permissions

The following table describes the minimum rights that must be applied to the Centrify Licenses container that stores the license keys for your installation.

This target object	Requires these permissions	Applied to
Licenses container	Read all properties Create classStore Objects Modify permissions	This object only
	Write Description property Write displayName property	This object and all child objects

The Setup Wizard requires you to create or select a parent container for license keys. The default location for the parent container for license keys depends on the organizational structure you deploy. For example, if you use the recommended organizational structure, the default location for licenses would be domain/Centrify/Licenses.

You must have at least one parent container for license keys in the forest. You might want to create more than one license container objects to give you more granular control over who has access to which licenses.

By default, all Authenticated Users have Read and List Contents permission for the Licenses container and all of its child objects. These permissions are required to use Access Manager. You can change who has these permissions if you want to prevent users from using Access Manager.

Zones Container Permissions

The following table describes the minimum rights that must be applied to the Centrify Zones container.

This target object	Requires these permissions	Applied to
Zones container or any container used as a destination for a new zone	Read all properties Create classStore Objects Create container objects	This object only
	Write displayName property	This object and all child objects
Change the default zone container	Delete	Previous zone container

The Setup Wizard requires you to create or select a parent Zones container object for new zones. The default location for the parent container for new zones depends on the organizational structure you deploy. For example, if you use the recommended organizational structure, the default location for new zones would be domain/Centrify/Zones. You can use other containers for zones or create multiple parent containers for zones to separate administrative duties for different groups.

Computers Container Permissions

The following table describes the minimum rights that must be applied to the generic Computers container (domain/Computers).

This target object	Requires these permissions	Applied to
Computers container	Write operatingSystem property Write operatingSystemVersion property Write operatingSystemHotfix property Write operatingSystemServicePack property	SELF on Computer objects

These permission are granted to each computer's SELF account when you select the **Grant computer accounts in the Computers container permission to update their own account information** option in the Setup Wizard.

Computers Container Within a Zone Permissions

The following table describes the minimum rights that must be applied to the Computers container in a named zone if you are supporting "agentless" authentication in that zone.

This target object	Requires these permissions	Applied to
ZoneName/Computers container	Create group objects Write Description property	This object only

Creating Parent Containers Manually

Some organizations prefer to create and manage Active Directory objects manually to ensure tight control over the objects and their related attributes. For example, you might want to manually create separate Zones or Licenses parent containers for different business units or geographic locations so that you can manually set different sets of permissions and related properties on those containers. Creating objects manually also enables you to have precise control over who has access to the objects.

You can create the container objects anywhere in the forest's directory structure, but you must have at least one parent Zones container object and at least one parent Licenses container object.

Optional Administrative Tasks

By default, Centrify does not require you to be an enterprise administrator or domain administrator of the forest root domain to install or configure Centrify-specific properties. However, some optional configuration tasks do require you to be an enterprise administrator or a domain administrator of the forest root domain.

These optional tasks involve:

- Creating display specifiers for Centrify profiles to enable access to the Centrify Profile properties page in Active Directory Users and Computers.
- Registering the administrative notification handler to ensure data consistency if users delete Centrify objects using Active Directory Users and Computers.
- Setting permissions for zones objects to enable maximum control over the placement of and rights associated with Centrify-related objects within Active Directory.

In most cases, if you want to perform any of these tasks, you must use an account that is an enterprise administrator or a domain administrator of the forest root domain.

Creating Display Specifiers for Centrify Profiles

To display the Centrify Profile properties in Active Directory Users and Computers, you must be an enterprise administrator or a domain administrator for the forest root domain because adding the Centrify Profile to Active Directory Users and Computers requires you to add display specifiers to Active Directory.

A display specifier is an Active Directory object that allows you to add components to the Active Directory Users and Computers (ADUC) Microsoft management console (MMC) snap-in.

If you want to make the Centrify Profile available in Active Directory Users and Computers, an enterprise administrator can manually define the display specifiers (under domain/Configuration/DisplaySpecifiers/LanguageID/) for computer, group, and user properties by modifying the adminPropertyPages attribute with the appropriate GUID. For example, if the Active Directory domain is ajax.org and the language you support is US-English (CN=409), you would define the display specifiers in:

ajax.org/Configuration/DisplaySpecifiers/409

Adding the display specifiers for Centrify properties is an optional step you can perform manually using ADSI Edit or by running the displayspecifier.vbs script. If you manage all Centrify objects through Access Manager, you do not need to perform this task.

To use the displayspecifier.vbs script to set up the display specifiers:

1. Log on using an enterprise administrator account or a domain administrator for the forest root domain.
2. Open a Command Prompt window and change to the Centrify installation directory. For example:

   cd C:\Program Files\Centrify\Access Manager
3. Run the displayspecifier.vbs script.

   If you want to manually add the display specifiers to display property pages in Active Directory Users and Computers, you must create the following entries using ADSI Edit, where n is the next number in the index of values for the attribute:

For this target object	Set this attribute	To
computer-Display displaySpecifier	adminPropertyPages	n,{DB5E4BE1-A0F0-4e6c-AD8A-B46475D727CB}
group-Display displaySpecifier	adminPropertyPages	n,{0CDC9AD0-E870-483f-8D16-17EAB3B7F881}
user-Display displaySpecifier	adminPropertyPages	n,{543DBFE3-317D-4493-8D00-84591E4EDCDE}
inetOrgPerson-Display	adminPropertyPages	n,{543DBFE3-317D-4493-8D00-84591E4EDCDE}

For example, if the Active Directory domain is ajax.org and the language you support is US-English (CN=409), you would add these entries to the objects in:

ajax.org/Configuration/DisplaySpecifiers/409

In most cases, you only need to set up the display specifiers once for the Active Directory forest. If you support multiple languages, you can manually add the display specifiers to each language you support. For example, if your organization supports US-English (CN=409), Standard French (CN=40C), and Japanese (CN=411), you would add the display specifiers to these three containers. Once you have updated Active Directory by running the displayspecifier.vbs script or by manually adding the display specifiers, you can access the Centrify Profile properties using Active Directory Users and Computers.

Registering the Administrative Notification Handler

The administrative notification handler provides services to ensure data integrity in the Active Directory forest. You can register the notification handler automatically through the Setup Wizard the first time you start Access Manager, but this requires an account that is an enterprise administrator or a domain administrator in the forest root domain.

Registering the administrative notification handler is optional, but doing so helps to ensure that no orphan UNIX data is left in the directory if a user, group, or computer is deleted using Active Directory Users and Computers. When registered, the notification handler automatically deletes any service connection point (SCP) dependencies on a directory object if the directory object is deleted. Without this service, deleting a directory object such as a computer or user account in Active Directory might leave an orphan service connection point for the object in the directory.

If you don't want to perform this step in the Setup Wizard, you can manually configure the administrative notification handler using ADSI Edit or you can choose not to register the administrative notification handler for Centrify. If you choose not to register the administrative notification handler, however, you should periodically run the Analyze command to check for orphan data in the Active Directory forest.

To manually set up the administrative notification handler for Centrify, add the following entry using ADSI Edit under domain/Configuration/DisplaySpecifiers/LanguageID/ where n is the next number in the index of values for the attribute:

For this target object	Set this attribute	To
DS-UI-Default-Settings	dSUIAdminNotification	n,{D0D2C2AE-C143-4C81-A61C-BE95C3C5EEDF}

For example, if the Active Directory domain is ajax.org and the language you support is US-English (CN=409), you would add this entry to the object in:

ajax.org/Configuration/DisplaySpecifiers/409

Granting Permissions For Administrative Tasks

The easiest way to grant permissions to perform administrative tasks is to use the Zone Delegation Wizard. The Zone Delegation Wizard enables you to delegate specific administrative tasks to specific users and groups. For each task you delegate to a specific user or group, you are providing that user or group with a specific set of permissions for working with objects in Active Directory.

The user who creates a zone has full control on the zone's serviceConnectionPoint. That user has exclusive permission to delegate administrative tasks to other users. The user who creates a zone is also the only user who can add NIS maps to the zone because creating NIS maps requires permission to create containers in Active Directory. The zone creator can, however, grant other users permission to add, remove, or modify NIS map entries.

The following table summarizes the permissions that can be assigned through your selections in the Zone Delegation Wizard. In addition to the permissions listed, however, the basic Read permission is required to perform any action. The Read permission is granted to Authenticated Users by default.

Selecting this task	Grants these rights
All	Permissions to perform all of the actions listed in the Zone Delegation Wizard and described below. This option allows a designated user or group to perform all of the other actions. Only the user who creates a zone can grant this permission to other users and groups for the zone.

Change zone properties	List contents on the ZoneName object container. Read all properties on the ZoneName object container. Write name on the ZoneName object container. Write Name on the ZoneName object container. Write Description property on the ZoneName object container.
Add users	List contents on the ZoneName/Users object container. Read all properties on the ZoneName/Users object container. Create serviceConnectionPoint objects on the ZoneName/Users object container.
Add groups	List contents on the ZoneName/Groups object container. Read all properties on the ZoneName/Groups object container. Create serviceConnectionPoint objects on the ZoneName/Groups object container.
Add local users	List contents on the ZoneName/Local Users object container. Read all properties on the ZoneName/Local Users object container. Add local users to the zone.
Add local groups	List contents on the ZoneName/Local Groups object container. Read all properties on the ZoneName/Local Groups object container. Add local groups to the zone.
Join computers to the zone	List contents on the ZoneName/Computers object container. Read all properties on the ZoneName/Computers object container. Create serviceConnectionPoint objects on the ZoneName/Computers object container. **Note** Joining the domain requires additional permissions on the Active Directory computer object, but the join command performs the necessary operations without requiring the additional permissions to be granted to the user or group you are designating as a trustee.
Remove zones	List contents on the ZoneName object container. Read all properties on the ZoneName object container. Allow Delete on the ZoneName object container. Allow Delete Subtree on the ZoneName object container.
Remove users	List contents on the ZoneName/Users object container. Read all properties on the ZoneName/Users object container. Delete serviceConnectionPoint objects on the ZoneName/Users object container.
Remove groups	List contents on the ZoneName/Groups object container. Read all properties on the ZoneName/Groups object container. Delete serviceConnectionPoint objects on the ZoneName/Groups object container.
Remove local users	List contents on the ZoneName/Local Users object container. Read all properties on the ZoneName/Local Users object container. Remove local users from the zone.
Remove local groups	List contents on the ZoneName/Local Groups object container. Read all properties on the ZoneName/Local Groups object container. Remove local groups from the zone.

Remove computers from the zone	List contents on the ZoneName/Computers object container. Read all properties on the ZoneName/Computers object container. Delete serviceConnectionPoint objects on the ZoneName/Computers object container.
Modify user profiles	List contents on the ZoneName/Users object container. Read all properties on the ZoneName/Users object container. Write cn on the serviceConnectionPoint object. Write name on the serviceConnectionPoint object. Write Name on the serviceConnectionPoint object. Write keywords on the serviceConnectionPoint object. For RFC 2307-compliant zones, modifying the user's UNIX profile also requires the following rights on the serviceConnectionPoint object of the UNIX user object: Write uid. Write uidNumber. Write loginShell. Write gidNumber. Write gecos. Write unixHomeDirectory. The additional rights for RFC 2307-compliant zones are applied to the posixAccount object associated with the serviceConnectionPoint for the UNIX user object.
Modify group profiles	List contents on the ZoneName/Groups object container. Read all properties on the object containers. Write name on the serviceConnectionPoint object. Write Name on the serviceConnectionPoint object. Write keywords on the serviceConnectionPoint object. For RFC 2307-compliant zones, modifying the group's UNIX profile also requires the following rights applied to the posixGroup object associated with the serviceConnectionPoint object of the UNIX group object: Write gidNumber.
Modify local user profiles	List contents on the ZoneName/Local Users object container. Read all properties on the ZoneName/Local Users object container. Modify local users in the zone. Parameters that can be modified are: User name (the UNIX login name). The user identifier (UID). The user's primary group profile numeric identifier (GID). The default home directory for the user. The default login shell for the user. General information about the user account (GECOS). State.
Modify local group profiles	List contents on the ZoneName/Local Groups object container. Read all properties on the object containers. Modify local groups in the zone. Parameters that can be modified are: Group name. Group members. Group identifier (GID). State.
Modify computer profiles	List contents on the ZoneName/Computers container object. Read all properties on the ZoneName/Computers container object. Write description on the ZoneName/Computers container object if the zone is a hierarchical zone. Write keywords on the serviceConnectionPoint object. Write displayName on the serviceConnectionPoint object. Write cn on the serviceConnectionPoint object. Write name on the serviceConnectionPoint object.
Allow computers to respond to NIS client requests	List contents on the ZoneName/Computers/zone_nis_servers group object. Read all properties on the ZoneName/Computers/zone_nis_servers group object. Write member property of group object on the ZoneName/Computers/zone_nis_servers group object.

Import users and groups to zone	List contents on the ZoneName/Users and ZoneName/Groups container object. Read all properties on the ZoneName/Groups container object. Create serviceConnectionPoint on the ZoneName/Users and ZoneName/Groups container objects. Write cn on the serviceConnectionPoint object. Write name on the serviceConnectionPoint object. Write managedby on the serviceConnectionPoint object. Write displayName on the serviceConnectionPoint object. Write keywords on the serviceConnectionPoint object. For RFC 2307-compliant zones, importing users also requires the following rights on the serviceConnectionPoint object of the UNIX user object ZoneName/Users: - Write uid. - Write uidNumber. - Write loginShell. - Write gidNumber. - Write unixHomeDirectory. - Write gecos. For RFC 2307-compliant zones, importing groups also requires the following right on the serviceConnectionPoint object of the UNIX group object under ZoneName/Groups: - Write gidNumber.
Manage roles and rights	List contents on the AzTask container and all child objects. Read all properties on the AzTask container and all child objects. Create msDS-AzTask objects Delete msDS-AzTask objects Write msDS-AzApplicationData on the msDs-AzTask object. Write cn on the msDs-AzTask object. Write name on the msDs-AzTask object. Write description on the msDs-AzTask object. Write msDs-OperationsForAzTask on the msDs-AzTask object. List contents on the AzOperation container and all child objects. Read all properties on the AzOperation container and all child objects. Create msDS-AzOperation objects Delete msDS-AzOperation objects Write msDs-AzApplicationData on the msDs-AzOperation object. Write cn on the msDs-AzOperation object. Write name on the msDs-AzOperation object. Write description on the msDs-AzOperation object. List contents on the msDS-AzAdminManager object. Read all properties on msDS-AzAdminManager object. Write msDs-AzApplicationData on msDS-AzAdminManager object.
Manage role assignments	List contents on the msDS-AzAdminManager object and all child objects. Read all properties on the msDS-AzAdminManager object and all child objects. Create msDS-AzRole objects. Delete msDS-AzRole objects. Write msDS-AzApplicationData on the msDS-AzRole object. Write msDS-TasksForAzRole on the msDS-AzRole object. Write msDS-MembersForAzRole on the msDS-AzRole object. Write displayName on the msDS-AzRole object. Write msDS-AzApplicationData on the msDS-AzAdminManager object.
Modify computer roles	List contents on the ZoneName object and all child objects. Read all properties on the ZoneName object and all child objects. Write msDS-AzApplicationData Write msDS-AzScopeName Write description
Add or remove NIS map entries	List contents on the ZoneName/NISMaps object container. Read all properties on the ZoneName/NISMaps object container. Create classStore Objects on the ZoneName/NISMaps object container. Write name on the ZoneName/NISMaps object container. Write Name on the ZoneName/NISMaps object container.

Modify NIS map entries	List contents on the ZoneName/NISMaps object container. Read all properties on the ZoneName/NISMaps object container. Write adminDescription on the classStore object. Write Description on the classStore object. Write wWWHomePage on the classStore object.
Remove NIS maps	List contents on the ZoneName/NISMaps object container. Read all properties on the ZoneName/NISMaps object container. Allow Delete on the MapName object. Allow Delete Subtree on the MapName object.

In some cases, the permissions granted through the Zone Delegation Wizard are a subset of the complete permissions required to perform some tasks. For information about the complete permissions required to perform a specific task, see the section that describes the permissions for performing that task. For example, for information about setting permissions for NIS maps, see Setting permissions for NIS maps.

Setting permissions for zones

The user who creates a zone has full control over zone properties and administrative tasks. Only the zone owner can delegate administrative tasks to other users and groups through the Zone Delegation Wizard. In most cases, the users who are allowed to create zones have domain administrator privileges and sufficient permissions to perform all administrative tasks and to delegate administrative tasks to other users.

If you manually set permissions to allow domain users to create zones, however, you should also manually set the permissions to allow those users to manage rights and roles or notify zone administrators that they should run the Zone Delegation Wizard and assign those tasks to their own account or to appropriate users and groups. At least one administrator must have permission to add an authorization store, define rights and roles, and manage role assignments in each zone. All users must have at least one valid role assignment to access a zone.

Creating a Zone

To create new zones, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Parent container for new zones you created or selected in the Setup Wizard. For example: domain/UNIX/Zones	On the **Object** tab, select **Allow** to apply the following permission to this object and all child objects: Create Container Objects Create Organizational Unit Objects **Note** Both permissions are required if you want to allow zones to be created as either container objects or organizational unit objects.
Parent container for Computers in the zone	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create group objects Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Write Description property These permissions are only needed if you are supporting “agentless” authentication in the new zone.

Opening Zones

To open an existing zone, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Parent container for new zones For example: domain/UNIX/Zones	On the **Object** tab, select **Allow** to apply the following permission to this object: List contents
Container for the individual zone For example, a ZoneName container object, such as: domain/UNIX/Zones/arcade	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributes Read allowedAttributesEffective Read canonicalName Read Description Read displayName Read name Read objectClass
Parent container for Users in the zone	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectClass
Parent container for Groups in the zone	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectClass

Modifying Zone Properties

To modify zone properties for a zone, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Container for an individual zone For example, a ZoneName container object: domain/UNIX/Zones/arcade	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read Name Read name Read Description Read displayName Write Description **Note** You can grant these permission to specific users or groups by selecting the **Change zone properties** task in the Zone Delegation Wizard. These permissions also enable you to change the parent zone for a selected zone object.

Renaming a Zone

To rename a zone, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Container for an individual zone For example, a ZoneName container object, such as: domain/UNIX/Zones/arcade	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Write name property Write Name property **Note** You can grant this permission to specific users or groups by selecting the **Change zone properties** task in the Zone Delegation Wizard.

Deleting a Zone

To delete a zone from Active Directory, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Container for an individual zone For example, a ZoneName container object, such as: domain/UNIX/Zones/arcade	On the **Object** tab, select **Allow** to apply the following properties to this object only: Delete Delete Subtree Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read Name Read name Read displayName **Note** You can grant this permission to specific users or groups by selecting the **Delete zone** task in the Zone Delegation Wizard.

Managing Roles and Rights in a Zone

To manage rights and roles in a zone, including creating and deleting role definitions and updating time constraints, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Container for the authorization store For example: domain/UNIX/Zones/arcade/Authorization	On the **Object** tab, select **Allow** to apply the following properties to this object and all child objects: List contents Read all properties Click the **Properties** tab and select **Allow** to apply the following properties to the msDS-AzAdminManager object: Write msDS-AzApplicationData
AzTaskObjectContainer	On the **Object** tab, select **Allow** to apply the following properties to this object and all child objects: List contents Read all properties Create msDS-AzTask objects Delete msDS-AzTask objects Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzTask objects: Write msDS-AzApplicationData Write cn Write name Write description Write msDs-OperationsForAzTask
AzOpObjectContainer	On the **Object** tab, select **Allow** to apply the following properties to this object and all child objects: List contents Read all properties Create msDS-AzOperation objects Delete msDS-AzOperation objects Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzOperation objects: Write msDS-AzApplicationData Write cn Write name Write description

Managing Role Assignments in a Zone

To manage role assignments in a zone, your user account must be set with the following permissions:

Select this target object	To apply these permissions
Container for the authorization store For example: domain/UNIX/Zones/arcade/Authorization	On the **Object** tab, select **Allow** to apply the following properties to this object only: List contents Read all properties Create all child objects Delete all child objects Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Write msDS-AzApplicationData Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzRole objects: Write displayName Write msDS-AzApplicationData Write msDS-TasksForAzRole Write msDS-MembersForAzRole
Computers container in the zone	On the **Object** tab, select **Allow** to apply the following properties to this object only: Create Container Right This permission is required to allow a delegated user to make the first role assignment after a computer is joined to Active Directory.
AzRoleObjectContainer	On the **Object** tab, select **Allow** to apply the following properties to the msDS-AzApplication object and all child objects: List contents Read all properties Create msDS-AzRole objects Delete msDS-AzRole objects Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzRole objects: Write displayName Write msDS-AzApplicationData Write msDS-TasksForAzRole Write msDS-MembersForAzRole Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzAdminManager objects: Write msDS-AzApplicationData
AzOpObjectContainer	On the **Object** tab, select **Allow** to apply the following properties to this object only: Read all properties Create msDS-AzOperation objects Delete msDS-AzOperation objects Create msDS-AzRole objects Delete msDS-AzRole objects Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzRole objects: Write displayName Write msDS-AzApplicationData Write msDS-TasksForAzRole Write msDS-MembersForAzRole Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzOperation objects: Read name Read Name Write msDS-AzApplicationData Write name Write description

Changing Computer Role Properties in a Zone

To manage computer role properties in a zone, your user account must be set with the following permissions:

Select this target object	To apply these permissions

Container for the authorization store For example: domain/UNIX/Zones/arcade/Authorization/guid The *guid* object is a globally unique identifier (GUID) for the Authorization object. For example: CN=cab186af-61a0-4d54-a0dd...	On the **Object** tab, select **Allow** to apply the following properties to this object only: Read all properties Click the **Properties** tab and select **Allow** to apply the following properties to msDS-AzScope objects: Read name Read Name Write msDS-AzApplicationData Write msDS-AzScopeName Write description

Setting Permissions to Join or Leave the Domain

To join a UNIX computer to an Active Directory domain without predefining a computer account, your Active Directory user account must be set with the following permissions:

Select this target object	To apply these permissions
Parent container object for computer accounts For example: domain/UNIX/Servers	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects **Note** You can grant this permission to specific users or groups by selecting the **Join computers** task in the Zone Delegation Wizard.

To join a UNIX computer to an Active Directory domain and place the computer account in a specific organizational unit (OU), the Active Directory account used to join the domain must be set with the following permissions:

Select this target object	To apply these permissions
Parent container object for the computer accounts	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects Create Computer Objects

To join a UNIX computer to an Active Directory domain when you are using a predefined computer account, your Active Directory user account must be set with the following permissions:

Select this target object	To apply these permissions
Parent container object for the computer account	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects
Computer account object in Active Directory For example, if the computer account is AJAX in the default Active Directory Computers container: domain/Computers/AJAX	On the **Object** tab, select **Allow** to apply the following permission to this object only: Full Control This permission is required for enabling or disabling a computer account.

To remove a UNIX computer from an Active Directory domain, your Active Directory user account must be set with the following permissions:

Select this target object	To apply these permissions
Parent container object for the computer account	On the **Object** tab, select **Allow** to apply the following permission to this object only: Delete serviceConnectionPoint Objects If you are deleting a computer account, you also need the Delete Computer Objects permission.

This setting only gives the user or group permission to leave an Active Directory domain. If you want to grant permission for a user or group to delete a computer account, you also need the Delete Computer Objects permission.

Setting Permissions for Zone Computers

Although joining or leaving a domain is the primary task for working with computer accounts in Active Directory, there are also specific permissions required to list computers or modify computer properties. The objects and permissions can also vary depending on the type of zone the computer account is associated with and the task to be performed. In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard.

In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard instead of assigning the permissions manually.

Joining a Computer to a Zone

To join a computer to a zone, your user account must have the following permission:

Select this target object	To apply these permissions
Parent container object for the computer account in the zone For example, in a classic zone, the ZoneName/Computers container object: domain/UNIX/Zones/acme/Computers	Click the **Object** tab and select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects
Computer account object in Active Directory For example, if the computer account name is AJAX: domain/UNIX/Servers/AJAX	Click the **Object** tab and select **Allow** for the Full Control permission for the user with permission to join the domain. The adjoin command grants the computer's SELF account the following permissions: Write operatingSystem Write operatingSystemVersion Write operatingSystemHotfix Write operatingSystemServicePack Write servicePrincipalName Write userAccountControl Write dnsHostName

Listing Computer Accounts

To list computers, your user account must have the following permission:

Select this target object	To apply these permissions
Parent container object for the computer account in Active Directory For example: domain/UNIX/Servers	On the **Object** tab, select **Allow** to apply the following permission to this object for each of the computers to be included in the list: List contents
Parent container object for the computer account in the zone For example, in a classic zone, the ZoneName/Computers container object: domain/UNIX/Zones/acme/Computers	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectClass
The serviceConnectionPoint object for the computer account For example, if the computer account name is AJAX, select: domain/UNIX/Servers/AJAX then select: serviceConnectionPoint objects	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each of the computers to be included in the list: Read displayName Read keywords Read managedBy Read Name Read objectClass
Computer account object in Active Directory For example, if the computer account name is AJAX: domain/UNIX/Servers/AJAX	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each of the computers to be included in the list: Read objectClass Read Operating System Read Operating System Version Read userAccountControl

Modifying Computer Properties

To modify any computer account properties for a UNIX computer, your user account must have the following permission:

Select this target object	To apply these permissions
Parent container object for the computer account in Active Directory For example: domain/UNIX/Servers	On the **Object** tab, select **Allow** to apply the following permission to this object only: List contents
The serviceConnectionPoint object for the computer account For example, if the computer account name is AJAX, select: domain/UNIX/Servers/AJAX then select: serviceConnectionPoint objects	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributes Read allowedAttributesEffective Read displayName Read keywords Read managedBy Read Name Read objectClass Write keywords
Computer account object in Active Directory For example, if the computer account is AJAX in the default Active Directory Computers container: domain/UNIX/Servers/AJAX	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectGUID Read objectSid Read objectClass Read Operating System Read Operating System Version Read userAccountControl

Responding to NIS Requests

If you are supporting “agentless” authentication or want to allow a computer to service NIS client requests in a zone, the computer must be a member of the zone_nis_servers group in the zone. Setting or unsetting the **Allow this computer to respond to NIS client requests** property requires the following permissions:

Select this target object	To apply these permissions
The zone_nis_servers group object For example, select: domain/UNIX/Zones/acme/Computers/zone_nis_servers	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: List contents Read all properties Write member property If the zone_nis_servers group does not already exist in the current zone, setting the **Allow this computer to respond to NIS client requests** property also requires the following permission on the ZoneName/Computers object: Create group objects

Changing the Computer Zone

If you need to change the zone for a computer account, your user account must have the following additional permissions:

Select this target object	To apply these permissions
All parent container objects for the original and new zones	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read name Read Name
The serviceConnectionPoint object for the computer account	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Write name Write Name **Note** The Name property is the common name (cn) of the serviceConnectionPoint object.
Original parent container for the computer account in the current zone For example, if you are moving a computer from the Finance zone to the Corporate zone, the target object would be: domain/UNIX/Zones/Finance/Computers	On the **Object** tab, select **Allow** to apply the following permission to this object only: Delete serviceConnectionPoint Objects Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectGUID
New parent container for the computer account in the new zone For example, if you are moving a computer from the Finance zone to the Corporate zone, and you use the default Computers container, the target object would be: domain/UNIX/Zones/Corporate/Computers	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectGUID

You can set the permissions for modifying computer accounts by clicking the **Security** tab when you are viewing a computer's properties.

Preparing a Computer Object

To prepare a computer account in a zone before joining, the following permissions apply to the user or group you want to designate as the trustee for joining the domain.

Select this target object	To apply these permissions
The serviceConnectionPoint object for the computer account	Click the **Object** tab and select **Allow** to apply the following permission to this object only: Read all properties Write keywords property Write displayName property
Computer account object in Active Directory For example, if the computer account name is AJAX: domain/Computers/AJAX	Click the **Object** tab and select **Allow** to apply the following permission to this object only: Read Permission Reset Password Write userAccountControl Validated write to DNS host name Validated write to service principal name Write to service principal name Write msDS-SupportedEncryptionTypes Write Account Restrictions Write Description Write displayName Write computer name (Pre-Windows 2000) Delete Delete Subtree All Extended Rights

The adjoin command resets the computer account and grants the computer's SELF account the following permissions:

- Write operatingSystem
- Write operatingSystemVersion
- Write operatingSystemHotfix
- Write operatingSystemServicePack
- Write altSecurityIdentities

Creating The Computer Object Manually

If you use Active Directory Users and Computers to prepare the computer object instead of the Prepare Computer wizard, the following permissions must be granted on the computer for the trustee:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the computer account	Click the **Object** tab and select **Allow** to apply the following permission to this object only: Read all properties Write keywords property Write displayName property

Computer account object in Active Directory For example, if the computer account name is AJAX: domain/Computers/AJAX	Click the **Object** tab and select **Allow** to apply the following permission to this object only: Read Permission Reset Password Write userAccountControl Validated write to DNS Host Name Validated write to service principal name Write Account Restrictions Write Description Write displayName Write computer name (Pre-Windows 2000) Write operatingSystem Write operatingSystemVersion Write operatingSystemHotfix Write operatingSystemServicePack Write altSecurityIdentities Write msDS-SupportedEncryptionTypes Delete Delete Subtree All Extended Rights

Modifying Computer Roles

If you use computer role assignments to control access to a computer, the following permissions are required to modify computer roles:

Select this target object	To apply these permissions
msDS-AzScope This object is listed under a globally unique identifier (GUID) for the Authorization object. For example: CN=cab186af-61a0-4d54-a0dd...	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read description Read msDS-AzScopeName Read msDS-AzApplicationData Write description Write msDS-AzScopeName Write msDS-AzApplicationData

Deleting Computer Roles

If you use computer role assignments to control access to a computer, the following permissions are required to delete computer roles:

Select this target object	To apply these permissions
msDS-AzScope This object is listed under a globally unique identifier (GUID) for the Authorization object.	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read Name Read name Read displayName Allow Delete Allow Delete Tree

Setting Permissions For Zone Users

The specific objects and permissions required to work with user accounts depend on the type of zone the user account is associated with and the task to be performed.

In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard instead of assigning the permissions manually.

Adding Users To Standard Zones

In a standard Centrify zone when the functional level of the forest is Windows Server 2003 or later, adding a user account with an Active Directory security group as the primary group to a zone requires the following permissions:

Select this target object	To apply these permissions
Parent container object for the user profile For example, if you use classic zones, the default Users container in the Finance zone: domain/UNIX/Zones/Finance/Users	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint Objects This permission is required for both standard zones and RFC 2307compliant zones. For standard zones, you need to apply additional permissions. Click the **Properties** tab and select **serviceConnectionPoint objects** from the object list, then select **Allow** to apply the following properties to this object: Read Name Read name Read displayName
User account object in Active Directory For example: domain/Users/user_name	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectCategory Read objectClass Read objectGUID Read objectSid Read userAccountControl
Parent container object for the individual zone For example, if you are adding a user to the Finance zone: domain/UNIX/Zones/Finance	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectGUID Write Description

Modifying Users In Standard Zones

In a standard zone, modifying user account properties for a user with a standard Active Directory security group as the primary group requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the user account For example, if you are using classic zones and the UNIX user name is chris: domain/UNIX/Zones/Finance/Users/chris then select serviceConnectionPoint objects	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributesEffective Read objectGUID Write keywords If you are changing the UNIX user name for the user, you need the following additional permissions applied to this object: Read name Write name Write Name property **Note** The Name property is the common name (cn) of the serviceConnectionPoint object.

You can set the permissions for modifying user accounts by clicking **Permissions** when you are viewing the Centrify Profile for a user.

Modifying Users In Rfc 2307-compliant Zones

In a standard RFC 2307-compliant zone, modifying user account properties for a user with an Active Directory security group as the primary group requires the following permissions:

Select this target object	To apply these permissions

The serviceConnectionPoint object for the user account For example, if you are using classic zones and the UNIX user name is chris: domain/UNIX/Zones/Finance/Users/chris then select serviceConnectionPoint objects	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributesEffective Write keywords Write uid Write uidNumber Write gidNumber Write loginShell Write unixHomeDirectory If you don't see some of these attributes listed for serviceConnectionPoint objects, change the object selected to **posixAccount objects**, then click **Allow** for the additional properties. The GECOS field in a user's UNIX profile is derived from the displayName attribute or the Name property (cn).

You can grant the required permissions to specific users or groups for any zone by selecting the **Modify users** task in the Zone Delegation Wizard.

Listing Users In Standard Zones

In a standard zone, listing user account information requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the user account	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each user included in the list: Read displayName Read managedBy Read objectClass Read Name to display the UNIX name Read keywords to display the other UNIX attributes

Listing Users in RFC 2307-Compliant Zones

In a standard RFC 2307-compliant zone, listing user account information requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the user account	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each user included in the list: Read displayName Read keywords Read managedBy Read objectClass Read uid to display the UNIX name Read uidNumber to display the UNIX UID Read gidNumber to display the GID of the user's primary group Read logonShell to display the default shell for the user Read unixHomeDirectory to display the user's home directory Read Public Information to display the userPrincipalName for the user

Removing Users from Zones

Removing a user account from a standard zone or RFC 2307-compliant zone requires the following permission:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the user account	On the **Object** tab, select **Allow** to apply the following permission to this object only: Delete

Setting Permissions for Zone Groups

The specific objects and permissions required to work with group accounts can vary depending on the type of zone the group is associated with and the task to be performed.

In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard instead of assigning the permissions manually.

Adding Security Groups to Zones

Adding an Active Directory group to a zone requires the following permissions:

Select this target object	To apply these permissions
Parent container object for the group For example, if you are using classic zones, the ZoneName/Groups container: domain/UNIX/Zones/acme/Groups	On the **Object** tab, select **Allow** to apply the following permission to this object only: Create serviceConnectionPoint objects Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectClass **Note** You can grant the required permissions to specific users or groups by selecting the **Add or remove groups** task in the Zone Delegation Wizard.
Group account object in Active Directory For example: domain/UNIX/UNIX groups/group_ name	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read groupType Read objectCategory Read objectClass Read objectGUID Read objectSid
Parent container object for the individual zone For example, if you are adding a group to the Finance zone: domain/UNIX/Zones/Finance	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectGUID Write Description

Modifying Groups in Standard Zones

In a standard zone, modifying a group profile in a zone requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the group account For example, if the UNIX group name is web-qa in the HKLab zone: domain/UNIX/Zones/HKLab/Groups/web-qa then select serviceConnectionPoint objects	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributesEffective Read objectGUID Read Name If you are changing the UNIX group name for a group, you need the following additional permissions applied to this object: Read name Write name Write Name **Note** The Name property is the common name (cn) of the serviceConnectionPoint object.

Modifying Groups in RFC 2307-Compliant Zones

In a standard RFC 2307-compliant zone, modifying a UNIX-enabled group in a zone requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the group account	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read allowedAttributesEffective Read objectGUID Read Name If you are changing the UNIX group identifier for a group, you need the following additional permissions applied to this object: Read gidNumber Write gidNumber **Note** If you don't see this attribute listed for the serviceConnectionPoint object, change the object selected to **posixGroup objects**. If you are changing the UNIX name for a group, you need the following additional permissions applied to this object: Read name Write name Write Name **Note** The Name property is the common name (cn) of the serviceConnectionPoint object.

Listing Groups in Zones

In a standard zone, listing group account information requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the group account	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each group included in the list: Read displayName Read managedBy Read objectClass Read Name to display the UNIX group name Read keywords to display the UNIX GID

Listing Groups in RFC 2307-Compliant Zones

In a standard RFC 2307-compliant zone, listing group account information requires the following permissions:

Select this target object	To apply these permissions
The serviceConnectionPoint object for the user account	Click the **Properties** tab and select **Allow** to apply the following properties to this object for each user included in the list: Read displayName Read keywords Read managedBy Read objectClass Read objectGUID Read Name to display the group name Read gidNumber to display the group GID

Removing Groups from Zones

Removing an Active Directory group from a standard zone or RFC 2307-compliant zone requires the following permission:

Select this target object	To apply these permissions

The serviceConnectionPoint object for the group account	On the **Object** tab, select **Allow** to apply the following permission to this object only: Delete

Setting Permissions for License Keys

Starting Access Manager requires the following permissions on the container object for licenses:

Select this target object	To apply these permissions
The domain root object For example, if the root domain of the forest is arcade.com: DC=arcade,DC=com	Click the **Properties** tab and select **Allow** to apply the following properties to this object only: Read objectClass
Parent container for the Licenses container object For example: domain/Centrify UNIX	On the **Object** tab, select **Allow** to apply the following permission to this object only: List contents
Parent container for license keys For example, the Licenses container object you created or selected in the Setup Wizard: domain/Centrify UNIX/Licenses	On the **Object** tab, select **Allow** to apply the following permission to this object only: List contents

To add and remove license keys, your user account must have the following permissions:

For this target object	You need these permissions
Parent container for license keys For example, the Licenses container object you created or selected in the Setup Wizard: domain/Centrify UNIX/Licenses	Click the **Properties** tab and select **Allow** to apply the following properties to this object and all child objects: Write Description

Setting Permissions for NIS Maps

You can delegate administrative permissions for all NIS maps in a zone or for any specific NIS map within a zone by selecting either the NIS Maps parent container object or the specific NIS map object you want to work with. If you select the NIS Maps parent container object, the permissions you set apply to all NIS maps you add to the zone. If you select the individual NIS map object, the permissions you set only apply to that individual NIS map.

To set permissions on NIS maps or NIS map entries

1. Open the ADSI Edit MMC snap-in and connect to the Active Directory domain.

   
   For NIS maps, you must use the Zone Delegation Wizard or ADSI Edit to set Active Directory permissions.

2. In the console tree, navigate to the zone folder.

   For example, if you deployed using the recommended organizational structure, expand the domain, Centrify, Zones, and select a specific zone name.

3. Select **CN=NisMaps** to set permissions for all NIS maps in a zone, right-click, then select **Properties**.

   If setting permissions for an individual map, expand CN=NisMaps, then select the map object–such as CN=auto_master–right-click and select **Properties**.

4. Click the Security tab, then click **Advanced**.
5. Click **Add** to search for the user or group to which you want to give administrative privileges, select the user or group in the results, then click **OK**.
6. Scroll to locate the appropriate permissions for the object and its properties to allow the selected user or group to perform the administrative task, click **Allow**, then click **OK**.

In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard instead of assigning the permissions manually.

Adding NIS Maps to a Zone

To add NIS maps to the NIS Maps parent container in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps For example, if you are using classic zones: domain/UNIX/Zones/ZoneName/NISMaps	On the **Object** tab, select **Allow** to apply the following permissions to this object and all child objects: Create Container Objects

Deleting NIS Maps from a Zone

To delete NIS maps in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps	On the **Object** tab, select **Allow** to apply the following permissions: Delete Container Objects applied to this object and all child objects. On the **Object** tab, set Apply onto to **Container objects**, then select **Allow** to apply the following permissions: Delete Subtree **Note** This permission is required if the map contains any entries.

Adding Map Entries to NIS Maps

To add entries to any NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps	On the **Object** tab, set Apply onto to **Container objects**, then select **Allow** to apply the following permissions: Create classStore Objects

Modifying Map Entries in NIS Maps

To modify entries in any NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps	Click the **Properties** tab, set Apply onto to **classStore objects**, then select **Allow** for the following properties: Write adminDescription Write Description Write wWWHomePage

Changing the Map Type for NIS Maps

To change the map type for any NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps	Click the **Properties** tab, set Apply onto to **This object and all child objects**, then select **Allow** for the following properties: Write Description

Deleting Map Entries from NIS Maps

To delete entries from any NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Parent container for NIS Maps	Click the **Properties** tab, set Apply onto to **classStore objects**, then select **Allow** for the following properties: Write name Write Name

Adding Entries to a Specific NIS Map

To add entries to a specific NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Individual NIS map	On the **Object** tab, select **Allow** to apply the following permissions to this object and all child objects: Create classStore Objects

Modifying Entries in a specific NIS Map

To modify the entries in a specific NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Individual NIS map	Click the **Properties** tab, set Apply onto to **classStore objects**, then select **Allow** for the following properties: Write adminDescription Write Description Write wWWHomePage

Changing the Map Type for a Specific NIS Map

To change the map type for a specific NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Individual NIS map	Click the **Properties** tab, set the Apply onto to **This object and all child objects**, then select **Allow** for the following properties: Write Description

Deleting Map Entries from a Specific NIS Map

To delete entries from a specific NIS map in a zone, the user account must have the following permissions:

Select this target object	To apply these permissions
Individual NIS map	Click the **Properties** tab, set Apply onto to **classStore objects**, then select **Allow** for the following properties: Write name Write Name

Setting Permissions for Password Synchronization

If you want to use the Network Information Service, adnisd, and the Centrify Password Filter to support “agentless” authentication of NIS client requests, the computer that will service the requests must be a member of the zone_nis_servers group in the zone and must be able to access the Active Directory attribute that stores the password hash. The specific permissions required depend on the attribute being used to store the password hash.

Centrify Password Synchronization Service

If you are using the Centrify Password Filter synchronization service, the zone_nis_servers group requires the following permissions:

If this attribute is used	These permissions are required
altSecurityIndentities	Read altSecurityIndentities
msSFU30Password	Read msSFU30Password
unixUserPassword	Read unixUserPassword All Extended Rights

Microsoft Password Synchronization Service

If you are using the Microsoft password synchronization service and the Centrify Network Information Service, adnisd, to authenticate NIS client requests, you must set the following permissions at the domain level, on the Users container object, or on another container that applies to all users.

Select this target object	To apply these permissions
Users container or a container that applies to all users	Click the **Object** tab, set the Apply onto to **User objects** and select **Allow** to apply the following permission: All Extended Rights You can apply this permission to Domain Computers or to a specific group of computers that contains the computer where the adnisd service is running.

For information about installing and configuring a Microsoft password synchronization service, see the Microsoft documentation for that service or refer to documentation on the Microsoft Web site.

Setting Permissions for Rights and Roles

If you define specific rights and establish role-based access controls on a zone-by-zone or computer-by-computer basis, you might want to manually assign permissions for users who can configure rights and roles.

In most cases, you can grant the required permissions to specific users or groups by selecting the appropriate task in the Zone Delegation Wizard instead of assigning the permissions manually.

Creating the Authorization Store

All of the information about rights, roles, and role assignments is held in an **authorization store** for each zone in Active Directory. The name of authorization store object is CN=Authorization under the zone object's DN. For example, the authorization store for the zone named EMEA_Territories in the Arcade.Net forest is:

cn=Authorization, cn=EMEA_Territories, cn=Zones, cn=UNIX, dc=Arcade, dc=Net

To create the authorization store for a zone, users must have the following permissions:

Select this target object	To apply these permissions
Parent container for an individual zone For example, a ZoneName container object, such as: domain/Centrify/Zones/arcade	On the **Object** tab, select **Allow** to apply the following permissions to this object and all child objects: List contents Read all properties Read Permissions Select **Allow** to apply the following permissions to this object only: Create msDS-AzAdminManager objects

Defining Rights And Roles in the Authorization Store

To configure rights, roles, and role assignments, users must have the following permissions for the authorization store:

Select this target object	To apply these permissions
Authorization	On the **Object** tab, select **Allow** to apply the following permissions to this object and all child objects: List contents Read all properties Write all properties
msDS-AzApplication This object is listed under a globally unique identifier (GUID) for the Authorization object. For example: CN=cab186af-61a0-4d54-a0dd...	On the **Object** tab, select **Allow** to apply the following permissions to this object (listed as CN=GUID under the Authorization object) and all child objects: Create and delete msDS-AzOperation objects Create and delete msDS-AzTask objects Create and delete msDS-AzRole objects Create msDS-AzScope objects **Note** You must grant these permissions on the CN=GUID object if you are granting permissions manually with ADSI Edit. The proper permissions are set automatically for the users when you delegate administrative tasks for a zone.

Configuring Authorization In Classic Zones

Unlike hierarchical zones, authorization is an optional feature in classic zones. You must be an administrator or the user who created a classic zone to initialize the authorization store in Active Directory, identify the users who should be allowed to configure rights, roles, and role assignments, and enable or disable the enforcement of the rights and role assignments you have configured.

To update the list of users and groups who are allowed to configure DirectAuthorize rights and roles, you must have the Modify permissions right on the Authorization container under the classic zone container applied to this object and all child objects. If you have this permission, you can click **Add** to add Windows users and groups to the list of users and groups who can configure rights and roles. If you have the Modify permissions right, you can also select a user or group in the list and click **Remove** a user or group from the list.

Adding Roles

To add roles for users or groups, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzTaskObjectContainer This object is listed under a globally unique identifier (GUID) for the Authorization object.	On the **Object** tab, select **Allow** to apply the following permissions to this object: Create msDS-AzTask objects Click the **Properties** tab, then select **Allow** for the following properties: Read objectClass

Modifying Roles

To modify roles for users or groups, users must have the following permissions:

Select this target object	To apply these permissions

Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzTaskObjectContainer/CN=roleName This object is listed under a globally unique identifier (GUID) for the Authorization object and a specific role name.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Read description Read msDS-AzApplicationData Write Name Write name Write description Write msDS-AzApplicationData

Deleting Roles

To delete roles for users or groups, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzTaskObjectContainer/CN=roleName This object is listed under a globally unique identifier (GUID) for the Authorization object and a specific role name.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Allow Delete

Adding Rights

To add the definition for a right in a zone, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-OpObjectContainer This object is listed under a globally unique identifier (GUID) for the Authorization object.	On the **Object** tab, select **Allow** to apply the following permissions to this object: Create msDS-AzOperation objects Click the **Properties** tab, then select **Allow** for the following properties: Read objectClass

Modifying Rights

To modify right definitions in a zone, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData

msDS-AzTaskObjectContainer/CN=roleName This object is listed under a globally unique identifier (GUID) for the Authorization object and a specific role name.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Read description Read msDS-AzApplicationData Write Name Write name Write description Write msDS-AzApplicationData

Deleting Rights

To delete right definitions in a zone, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzOpObjectContainer/CN=pamrightName or msDS-AzOpObjectContainer/CN=pcrightName This object is listed under a globally unique identifier (GUID) for the Authorization object and a specific PAM access right name or privileged command name.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Allow Delete

Adding or Removing Rights from Roles

To add or remove rights from a role in a zone, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzTaskObjectContainer/CN=roleName This object is listed under a globally unique identifier (GUID) for the Authorization object and a specific role name.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Read msDS-OperationsForAzTask Write msDS-OperationsForAzTask

Adding Role Assignments

To add a role assignment, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData

msDS-AzRoleObjectContainer This object is listed under a globally unique identifier (GUID) for the Authorization object.	On the **Object** tab, select **Allow** to apply the following permissions to this object: Create msDS-AzRole objects

Modifying Role Assignments

To modify role assignments, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzRoleObjectContainer This object is listed under a globally unique identifier (GUID) for the Authorization object.	On the **Object** tab, select **Allow** to apply the following permissions to this object: Create msDS-AzRole objects
msDS-AzRoleObjectContainer/CN=CRA_guid This object is listed under a globally unique identifier (GUID) for the Authorization object and a unique identifier for the role assignment.	Click the **Properties** tab, then select **Allow** for the following properties to allow changes to the assigned user or groups: Read Name Read name Allow Delete Click the **Properties** tab, then select **Allow** for the following properties to allow changes to the available time for a role assignment: Read Name Read name Read msDS-AzApplicationData Write msDS-AzApplicationData

Deleting Role Assignments

To modify role assignments, users must have the following permissions:

Select this target object	To apply these permissions
Authorization	Click the **Properties** tab, then select **Allow** for the following properties: Write msDS-AzApplicationData
msDS-AzRoleObjectContainer/CN=CRA_guid This object is listed under a globally unique identifier (GUID) for the Authorization object and a unique identifier for the role assignment.	Click the **Properties** tab, then select **Allow** for the following properties: Read Name Read name Allow Delete

Setting Permissions for Zone Provisioning

The Zone Provisioning Agent requires permission to create UNIX profiles, that is, the service connection points in each zone where it needs to perform provisioning operations. The service account that runs the Zone Provisioning Agent requires the Log on as a service right set as a local computer security policy, or in the default domain policy.

Supplemental Installation Notes

This document includes various notes about installing Server Suite on different operating system platforms.

Verifying the DNS Configuration on Linux

The Server Suite Authentication Service (DirectControl) uses DNS to locate domain controllers for the Active Directory forest. To verify the Active Directory domain controller can be located through DNS, try sending a ping request to the computer.

You can also run the adinfo --diag command to attempt to read the DNS records for the domain you want to join. For example:

adinfo --diag domain_name

If DNS is properly configured, the command should display the LDAP URLs for the domain controllers in the domain you want to join.

For more detailed information about configuring DNS or troubleshooting your DNS configuration, see the *Administrator's Guide for Linux and UNIX.*

Joining the Domain (Zoned Mode Only)

To join an Active Directory domain manually:

1. On the Linux computer, log in as or switch to the root user.
2. Run adjoin to join an existing Active Directory domain using a fully-qualified domain name.

   adjoin --zone <zone_name> --user <user_name><domain_name>

   The user account you specify must have permission to add computers to the specified domain and zone. If you don't specify a user name, the Administrator account is used by default.
3. Type the password for the specified user account.

   If the authentication service can connect to Active Directory and join the domain, a confirmation message is displayed. You can now enable existing Active Directory groups and users to work with this Unix computer.

For more information about the options you can specify when joining a domain, see the man page for the adjoin command or the *Administrator's Guide for Linux and UNIX.*

To step through common tasks and test scenarios, see the *Evaluation Guide for Linux and UNIX*.

Joining the Domain (Express mode)

To join an Active Directory domain manually:

1. On the UNIX computer, log in as or switch to the root user.
2. Run adjoin to join an existing Active Directory domain using a fully-qualified domain name.

   adjoin --workstation --user <user_name> <domain_name>

   The user account you specify must have permission to add computers to the specified domain. If you don't specify a user name, the Administrator account is used by default.

3. Type the password for the specified user account.

   If the authentication service can connect to Active Directory and join the domain, a confirmation message is displayed.

   For more information about the options you can specify when joining a domain, see the man page for the adjoin command or the *Administrator's Guide for Linux and UNIX*.

HPUX Installation Notes

This section describes the unique characteristics or known limitations that are specific to using authentication service on a computer with the HP-UX operating environment.

ia64 - Mapping Local HP-UX User Accounts to Active Directory Accounts

In most environments, you can map local user accounts to Active Directory accounts to manage the passwords for local users using your Active Directory password policies. On HP-UX, however, if an account is a valid Active Directory account but the authentication through Active Directory fails, the PAM module will attempt to authenticate the account locally and will allow the account to log on if the local authentication succeeds. Because users can still log on to HP-UX systems using their local account password, you cannot effectively use Active Directory or the User Map group policy to enforce your password policies for local HP-UX user accounts.

To enforce Active Directory password policies for local HP-UX users, you need to disable the local user accounts to prevent those local account names and passwords from being used to log on.

Entering an Incorrect Password on HP-UX

On HP-UX, if Server Suite-enabled users enter an incorrect password, they are normally prompted with a second "System password" prompt. This prompt is asking for a password for a local user, regardless of whether that user actually exists locally on the system. If the user exists locally, this prompt allows the user to log in using the local password. If the user does not exist locally, this prompt is unnecessary and will not allow the authentication service-enabled user to log in, regardless of the password entered.

This second prompt can be avoided by changing the options in /etc/pam.conf to the authentication modules. Two changes are necessary:

1. Add an option to the authentication service PAM module to prompt all users for a password (not just Active Directory users)
2. Add an option to the HP-UX UNIX login module to use the password obtained by the authentication service module.

   The lines which need to be modified appear like this in the file:

   service_name auth sufficient /usr/lib/security/libpam_centrifydc.1 debug

   service_name auth required /usr/lib/security/libpam_unix.1

   Where service_name is something like login, dtlogin, ftp, or similar. The pam_centrifydc.1 line needs the ask flag to prompt all users for passwords. The libpam_unix.1 line needs the use_first_pass option. For example:

   login auth sufficient /usr/lib/security/libpam_centrifydc.1 debug ask

   login auth required /usr/lib/security/libpam_unix.1 use_first_pass

It is extremely important that the pam_centrifydc line appear before the pam_unix line in the file, or users will never be prompted for a password. Administrators should be extremely careful when editing this file. Any typographically errors in this file could prevent all users from logging on to the system and render the system unusable.

AIX Installation Notes

Support for AIX Capabilities Attribute

Support has been added for the AIX Capabilities user attribute, a feature that is only available on AIX 5.3 and later. To enable the feature, edit /etc/centrifydc/centrifydc.conf to add the following line:

lam.method.version: 520

This allows using methods that are only available with AIX 5.3 and later, and these methods are required to support the Capabilities attribute.

Use adquery to view capabilities for an Active Directory user:

adquery user -X aix.capabilities <ADuser>

Use adupdate to set capabilities for an Active Directory user:

adupdate modify user -X +aix.capabilities=CAPABILITIES <ADuser>

Where CAPABILITIES is a comma-separated list of capabilities to add for the user. For example:

CAP_NUMA_ATTACH, CAP_BYPASS_RAC_VMM, CAP_PROPOGATE

Currently there is no group policy support for capabilities, this may be implemented in a future release of authentication service.

Users Cannot Log in by way of FTP if They Have a Restricted Shell

On AIX 6.1, a user's login shell must appear in the shells attribute of the /etc/security/login.cfg file. Delinea Privilege Elevation Service does not add dzsh to this attribute so by default an ftp user who is using dzsh as their login shell cannot log in. To workaround this issue, add /usr/bin/dzsh to the shells attribute of /etc/security/login.cfg.

Starting and Stopping DirectControl on AIX

Because the authentication service daemon, adclient, is defined as an AIX system resource, you use the following commands to start, stop, and check the status of the daemon:

startsrc -s centrifydc

stopsrc -s centrifydc

lssrc -s centrifydc

Using the Server Suite Authentication Service LDAP Proxy on AIX

When using the LDAP Proxy on AIX you need the following line in the slapd configuration file at

/usr/share/centrifydc/etc/openldap/ldapproxy.slapd.conf
moduleload /usr/share/centrifydc/libexec/openldap/libback_centrifydc.a(libback_centrifydc.so.0)

This should be entered as a single line into the configuration file. This line may already be in the configuration file, but commented out, in which case you can just remove the leading "#" to uncomment it.

Setting the DNS Configuration Parameter to Join the Domain on SuSE Linux

To successfully join a Active Directory domain on computers running SuSE Linux, you must set the mdns option to off in the /etc/host.conf file. If your /etc/host.conf file does not include the following line, you should add it to the file:

mdns off

This setting is required to enable proper DNS resolution, and therefore, must be set to successfully join the domain, and to allow users to log on properly.

Mounting CIFS Shares

Common Internet File Systems (CIFS) provides an open and cross-platform protocol for requesting remote network server files and services. When a CIFS share is mounted on a Centrify Linux system, file ownership is listed incorrectly.

To correct this, apply the CentrifyDC-cifsidmap plug-in. The CentrifyDC-cifsidmap plug-in enables mapping AD User/Group Security IDs (SIDs) to User/Group IDs (UIDs/GIDs) configured in a zone and from UIDs/GIDs to AD User/Group SIDs correctly. This, in turn, allows the CIFS Client integration with DirectControl.

Use Cases

Mapping UIDs to SIDs is not always required when mounting CIFS shares. But it is needed when working with the files on the shares. For example, when modifying Access Control Links (ACLs). In version 5.8 and older, the cifs-utils package uses the winbind daemon for this mapping. Through winbind, the /usr/sbin/cifs.idmap binary was linked against libraries.

The /usr/sbin/cifs.idmap binary works in conjunction with the Samba winbind facility to map owner and group SIDs to UIDs and GIDs respectively.

With version 5.9 the winbind facility does not perform this mapping. Use the **CentrifyDC-cifsidmap** plug-in to ensure that:

- cifs-idmap translates the ownership on the SMB share correctly.
- the kernel determines who has rights to the CIFS share mount directories and files correctly.
- AD User/Group SIDs are mapped correctly and all the IDs are consistent and correct.

For example:

To see the incorrect file ownership: mount your CIFS share and display the ownership of the files in the mounted share.

1. Mount the share. This command requires root privileges.

   Syntax:

sudo mount -t cifs *domain_ip/path/local/path/* -o username=*your_user_name*, *file_privilege*, password=*your_password*, domain=*domain_name*, cifsacl

Example:

sudo mount -t cifs //192.168.0.100/cifsshare /tmp/mntshare1/ -o username=cifsdemouser1,rw, password=My1Pass,domain=example.com,cifsacl

The cifs type (-t cifs) requires the cifsacl option. See man mount.cifs for command usage.

2. List all the files on the mounted file system.

   If the CIFS share is owned by root, then you need to use sudo to view the files on the mounted directory, because the files you are verifying can only be seen with root privileges.

   sudo ls -al /mntshare1

   ... ... root root ... cifsdemouser1.txt
   ... root root ... cifsdemouser2.txt
   ... root root ... cifsdemouser3.txt

   If the AD user names are not listed, and only root is listed at the owner of the files, then you need to install the CentrifyDC-cifsidmap plug-in. Complete the steps in the following sections.

CentrifyDC-cifsidmap Plug-in Requirements

The Centrify CIFS idmap plug-in is available only for supported systems. The cifs.idmap-plugin requires:

Operating system versions:

- RedHat 7 or above
- Debian 8 or above
- SUSE 12 or above

cifs-utils version:

- cifs-util 5.9 or above

Prepare to Install the CentrifyDC-cifsidmap Plug-in

Prior to installing the CentrifyDC-cifsidmap plug-in, install and configure the following:

- Install the cifs-utils

  The cifs-utils are a package of tools used on CIFS filesystems. See your CIFS documentation.

- Install CentrifyDC

  See the Planning and Deployment Guide.

- Join the machine to AD

  See the Planning and Deployment Guide.

Install the CentrifyDC-cifsidmap Package

1. Verify cifs-utils package is installed. Install it, if it is not already installed. For example:

   It is possible to manually configure your system without cifs-utils, but the program /usr/sbin/cifs.idmap, is still required for the Centrify CIFS idmap plug-in to work.

   - SUSE or RedHat

     yum install cifs-utils

   - Debian

     apt-get install cifs-utils

2. Download the CentrifyDC-cifsidmap package and change to the download directory.

   The package contains the cifs-idmap-plugin.

   Example download package names

   - SUSE or RedHat

     CentrifyDC-cifsidmap-5.5.1-rhel5.x86_64.rpm

   - Debian

     CentrifyDC-cifsidmap-5.5.1-deb8-x86_64.deb

     Example download directory

     # cd /home/user1/Download/

3. Run the appropriate package install command from the download directory.

   - SUSE or RedHat

     # sudo rpm -i CentrifyDC-cifsidmap-5.5.1-rhel5.x86_64.rpm

   - Debian

     # sudo dpkg -i CentrifyDC-cifsidmap-5.5.1-deb8-x86_64.deb

4. Verify the CentrifyDC-cifsidmap package is installed correctly. Check the libcifsidmap.so location.

   - SUSE or RedHat

     # ls /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so -al

     ... /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so

   - Debian

     # ls /usr/share/centrifydc/lib/plugins/cifs/libcifsidmap.so -al

     ... /usr/share/centrifydc/lib/plugins/cifs/libcifsidmap.so

Configure cifs-utils for CentrifyDC-cifsidmap Plug-in

On Linux, the command, alternatives, is a tool for managing different software packages that provide the same functionality. The alternatives command, on different systems has different names and locations. For additional information on alternatives use, see your Linux documentation.

- RedHat

  /usr/sbin/alternatives

- SUSE and Debian

  /usr/sbin/update-alternatives

To configure the cifs-utils

1. Check the status of /etc/cifs-utils/idmap-plugin and note the priority level.

   For example on RedHat:

   # pwd

   /etc/cifs-utils

   # ls -al

   ... idmap-plugin -- /etc/alternatives/cifs-idmap-plugin

   # alternatives --display cifs-idmap-plugin

   ...
   /usr/lib64/cifs-utils/cifs_idmap_sss.so - priority 20
   ... Current 'best' version is /usr/lib64/cifs-utils/cifs_idmap_sss.so.

   In this example the cifs_idmap_sss.so plugin object has the highest priority and that priority is set to 20.

2. Configure cifs-utils to use the CentrifyDC-cifsidmap plug-in, cifs-idmap-plugin.

   Run the commands appropriate for your OS.

   Include a priority that is higher than the priority listed in Step 1. For example, the priority in Step 1 is 20, set this cifs-idmap-plugin priority to 21 or higher.

   - RedHat

     alternatives --install /etc/cifs-utils/idmap-plugin cifs-idmap-plugin
     /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so <*priority*>

     alternatives --set cifs-idmap-plugin /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so

   - SUSE or Debian

     update-alternatives --install /etc/cifs-utils/idmap-plugin cifs-idmap-plugin
     /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so <*priority*>

     update-alternatives --set cifs-idmap-plugin /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so

3. Verify the CentrifyDC-cifsidmap plug-in is configured correctly. Run the appropriate alternatives display option.

   - RedHat

     alternatives --display cifs-idmap-plugin

   - SUSE or Debian

     update-alternatives --display cifs-idmap-plugin

4. Verify the cifs-idmap-plugin location and priority. Review the alternative command response.

The cifs-idmap-plugin priority needs to be higher than other listed idmaps. The Current 'best' version needs to point to the cifs-idmap-plugin location.

For example on RedHat:

```
# alternatives --display cifs-idmap-plugin
... /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so - priority 21
 Current 'best' version is /usr/share/centrifydc/lib64/plugins/cifs/libcifsidmap.so.
```

Mount the CIFS Share and Confirm File Ownership

Only mount CIFS shares as root user or use sudo.

1. Verify the receiving mount directory. Create a directory to receive the mount files, if you do not have one. For example:

   ```
   cd /tmp
    mkdir mntshare1
    ls -al /mntshare1
   ```

2. Optionally, verify that the user(s), you are expecting to be owners of CIFS shared files, are valid AD users. For example:

   ```
   adquery user cifsdemouser1
   ```

   ```
   cifsdemouser1:x:1019226236:1019226232:cifsdemouser1:home/cifsdemouser1:/bin/bash
   ```

3. If you previously mounted the CIFS share, and found that file ownership was incorrect, unmount it now. For example:

   ```
   sudo unmount /tmp/mntshare1/
   ```

4. Mount the share. This command requires root privileges.

   Syntax:

   ```
   sudo mount -t cifs domain_ip/path/local/path/ -o username=your_user_name, file_privilege,
    password=your_password, domain=domain_name, cifsacl
   ```

   Example:

   ```
   sudo mount -t cifs //192.168.0.100/cifsshare /tmp/mntshare1/ -o username=cifsdemouser1,rw,
    password=My1Pass,domain=example.com,cifsacl
   ```

   The cifs type (-t cifs) requires the cifsacl option. See man mount.cifs for command usage.

5. List all the files on the mounted file system.

   If the CIFS share is owned by root, then you need to use sudo to view the files on the mounted directory, because the files you are verifying can only be seen with root privileges.

   ```
   sudo ls -al /tmp/mntshare1
   ```

   ```
   ... ... cifsdemouser1 root ... cifsdemouser1.txt
    ... cifsdemouser2 root ... cifsdemouser2.txt
    ... cifsdemouser3 root ... cifsdemouser3.txt
   ```

   Notice the AD users are listed as owners of the CIFS share files. This completes the task.

Known Issues

Here are some known issues, organized by category.

Installation and Un-installation Issues

- Upgrading from the beta build to this version may result in offline MFA mode if there are multiple authentication servers registered in your AD forest. To resolve this, uninstall the beta build first and then install this new version. (Ref: CS-41915)
- The Centrify Common Component should be the last Server Suite component uninstalled. If the component is uninstalled before other component, it must be reinstalled by the uninstall process to complete its task. (Ref: 36226a)
- If you intend to install the software on the desktop with elevated privilege, you should not check the “Run with UAC restrictions” option when creating the desktop. (Ref: 39725b)
- When you double-click on the Server Suite Agent for Windows msi and select the “repair” option, the existing files are replaced irrespective of their version number, even when they are identical. As a result, a prompt to restart the system is displayed as files that were in use were replaced. However, if you use the Easy Installer to do the repair and a file on the disk has the same version as the file that is part of the installer package, the installed file will not be replaced. Therefore, there will not be any prompt to restart the system. (Ref: 26561a)
- If you uninstall the Server Suite Agent for Windows while the DirectAudit Agent Control Panel is open, files needed by the uninstall process may be blocked. You should close the DirectAudit Agent Control Panel for a successful conclusion to the uninstall process. (Ref: 25753a)
- Server Suite Agent for Windows and its installer are built on .NET. Therefore, .NET is always installed as a pre-requisite before the agent is installed. If .NET is removed from the system later, Server Suite Agent for Windows will not run properly. User will also experience problem when trying to remove Server Suite Agent for Windows from the system. To properly uninstall Server Suite Agent for Windows, please make sure Server Suite Agent for Windows is uninstalled before .NET. (Ref: 39051a)

Configuration Issues

- In a cross-forest environment, forest A user cannot enroll a device joined to forest B when forest A does not have a connector. (Ref: CS-44805)
- In Windows 2016 and Windows 10, during the login process, selecting SMS or using other mechanisms like Security Question/Phone call/Password/Email/Mobile for MFA and clicking the “Commit” button will be intermittently unresponsive. (Ref: CS-41699)
- In some large environment with multiple domain controllers, it may take up to one minute for the new zone setting in Server Suite Agent Configuration to take effect. (Ref: 58128b)
- If one of the Global Catalog servers is unavailable, user may not be able to configure the zone for Server Suite Agent for Windows. (Ref: 58621b)
- Microsoft normally automatically distributes and installs root certificates to the Windows system from trusted Certificate Authorities (CA) and users are seamlessly able to use a secure connection by trusting a certificate chain issued from the trusted CA. However, this mechanism may fail if the system is in a disconnected environment where access to Windows Update is blocked or this feature of automatic root certificate installation is disabled. Without updates on the certificate trust list (CTL), the default CTLs on the system may not be

adequate for secure connections of multi-factor authentication especially for older versions of Windows such as Windows 7 and Windows Server 2008 R2. To ensure the success of multi-factor authentication, user may need manually distribute and install the latest CTLs and the required root certificate to systems in a disconnected environment. See Centrify KB-6724 for further information. (Ref: CS-39703)

Environment Issues

- On Windows 10 and Windows 2016 machines with Centrify Privilege Elevation Service, the following will occur (Ref: CS-43883):
  - Pop up an error dialog several seconds after clicking "Open file location" in the context menu of a shortcut on the start menu. Explorer windows will display correctly.
  - No responses to the following actions
    - Clicking "Open file location" in the context menu of a shortcut on desktop
    - Clicking "Open file location" in the context menu of a shortcut on the Centrify Start menu in the Privileged Desktop
    - Slow response to "OK", "Cancel" in the shortcut property page after "Open file location" in the general tab is clicked. The dialog will close after several seconds.
- On some Windows 10 computers, the smart card login option may not be displayed if another credential method has been recently used. To display the smart card login option, remove and insert a smart card into the reader. This will cause the login screen to reload and will display the smart card login option. (Ref: CS-41282)
- An environment with no Global Catalog is not supported. (Ref: 46577a)
- Centrify Privilege Elevation Service requires machine time to be synchronized with domain controller. VMware virtual machine has a known issue that its time may not be synchronized with domain controller. This problem occurs more often on an overloaded virtual machine host. If the system clocks on the local Windows computer and the domain controller are not synchronized, Centrify Privilege Elevation Service does not allow any domain users to login. You can try the following KB from VMware to fix the time synchronization issue. http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1189 (Ref: 47795b)

RunAsRole Issues

- If you use the “RunAsRole.exe /wait” command to run a Python script, the input/output cannot be redirected for versions of Python below 3.0.0. (Ref: 45061a)
- The Run As Role menu is not available on the start screen in Windows 8 or Windows 2012 or later because Microsoft doesn’t support any custom context menu on the start screen. User has to go to the Windows desktop in order to launch an application using Run As Role context menu. (Ref: 35487a)
- When running “RunAsRole.exe /wait sc.exe” with no argument provided to sc.exe, sc.exe will prompt
  - Would you like to see help for the QUERY and QUERYEX commands? [y | n]:
  - Typing ‘y’ or ‘n’ doesn’t do anything because the input cannot be successfully redirected to sc.exe. (Ref: 47016b)
- It is not recommended to change zone via Run As Role since the role that is in use may no longer be available once after leaving from the previous zone during the change zone process. (Ref: 58043a)

- On Windows Server 2008 R2 and Windows 7, if the Agent machine has no internet connection and the .NET CLR settings (checkCertificateRevocationList) is set to True, the MFA authentication will be failed because the CLR is unable to verify the certificate through internet. The workaround is to enable the internet connection or turn off the CLR settings (set checkCertificateRevocationList to False which is also the default value). (Ref: CS-40147)

Desktop with Elevated Privileges issues

- On a desktop with elevated privileges, if you use “Control Panel > Programs > Programs and Features” to uninstall a program, you may see the following warning message and cannot uninstall the software.

  “The system administrator has set policies to prevent this installation.”

  This issue happens when User Account Control (UAC) is enabled and when “Run with UAC restrictions” is selected when creating the new desktop. (Ref: 33384a)
- You cannot use the Start menu option “Switch User” while you are using a role-based, privileged desktop. To use the “Switch User” shortcut, change from the privileged desktop to your default Windows desktop. From the default desktop, you can then select Start > Switch User to log on as a different user. (Ref: 39011b)

Roles and Rights Issues

- There is no 'Require multi-factor authentication' system right for the predefined 'Windows Login' role. To define this system right for MFA, use the pre-defined Require MFA for logon role, or create a new custom role. (Ref: CS-40888)
- Windows Network Access rights do not take effect on a Linux or UNIX machines. If you select a role to start a program or create a desktop that contains a Network Access right, you can only use that role to access Windows computers. The Windows computers you access over the network must be joined to a zone that honors the selected role. The selected role cannot be used to access any Linux or UNIX server computers on the network. (Ref: 32980a)
- Network Access rights are not supported on the Windows 2008 R2 Terminal Server if “RDC Client Single Sign-On for Remote Desktop Services” is enabled on the client side. (Ref: 34368b)
- To elevate privileges to the “Run as” account specified in a Windows right, the “run as” account must have local logon rights. If you have explicitly disallowed this right, you may receive an error such as “the user has not been granted the requested logon type at this computer” when attempting to use the right. (Ref: 34266a)
- If your computer network is spread out geographically, there may be failures in NETBIOS name translation. If a NETBIOS name is used, Active Directory attempts to resolve the NETBIOS name based on the domain controller that the user belongs to, which in a multi-segment network might fail. Therefore, Network Access rights might not work as expected if the remote server is located using NETBIOS name. You may need to consult your network administrator to work around this issue. (Ref: 39087a)
- File hash matching criteria in the Application right is not supported for a file larger than 500MB. This is to make sure DirectAuthorize does not spend too much CPU and memory resources to calculate the file hash. User trying to import a file with the size larger than 500MB will see an empty value for the file hash field. (Ref: 56778a)
- For a small set of application, enabled matching criterion - “Product Name”, “Product version”, “Company”, “File Version” or “File Description” of a Windows Application Right may fail to match after upgrading agent under the following conditions: - Any value for the enabled matching criteria is defined by either import from a process or file - The matching criteria is defined by 5.1.3 or 5.2.0 DirectManage Access Manager since the number of

affected application is expected to be relatively low, proactively updating the defined matching criteria of Windows Application Right is not necessary. (Ref: 60053a)

Compatibility with Third Party Products Issues

- VirtualDesktop is not compatible with Server Suite Agent for Windows. Users should use the Centrify system tray applet to create virtual desktop instead. (Ref: 44641b)
- The startup path for “SharePoint 2010 Management Shell” and “Exchange Management Shell” may set to C:\Windows instead of user home directory if it is launched via RunAsRole.exe or from a desktop with elevated privilege. (Ref: 38814b, 46943b)
- Attempting to enable Kerberos authentication for Oracle databases will fail. This issue is being brought to the attention of Oracle Support for a resolution in upcoming releases. (Ref: 33835b)
- Some applications do not use the process token to check the group membership. They check the user’s group membership on its own. Therefore, any Windows rights configured to use a privileged group will not take effect in these applications. The workaround is to use a privileged user account instead of a privileged group. Here is the list of known application with this issue:
  - vCenter Server 5.1
  - SQL Server
  - Exchange 2010 or above
  - SCOM 2007

  (Ref: 45318a, 45218a, 43779a, 38016a)
- Users may notice an error and cannot install ActivClient after installing Server Suite Agent for Windows. During the installation of ActivClient, it attempts to change the local security setting. However, there is a known issue for Server Suite Agent for Windows of blocking the local security setting (Ref: 63609b). Therefore, users may not be able to install ActivClient successfully after installing Server Suite Agent for Windows. We suggest installing ActivClient before installing Server Suite Agent for Windows. If Server Suite Agent for Windows has been installed, please uninstall it and follow the installation sequence suggested. This issue happens on Windows 8.1 and Windows 2012 R2 only. (Ref: 76016b)

Application Manager Issues

Application Manager does not support the Server Core edition of Windows. (Ref: CS-40656)

Best Practices

This section, created by Delinea Systems Engineering in collaboration with Delinea Engineering and Delinea Professional Services, describes the deployment best practices for Server Suite. The goal of this document is to outline and document the actions customers can take to prevent unexpected service degradation with the Server Suite product.

Using our best practices that have evolved over many years, we have developed the Server Suite software to be extremely resilient to many types of Active Directory topologies, networks and environments. In addition, we have gathered data from our major deployments to provide the top items that a customer should do to ensure the Server Suite deployment is healthy. The best practices are organized by functional area.

Best Practices For Unix And Linux Systems With Server Suite

This section includes the following topics:

Upgrade Server Suite Agents And Administrative Tools

Many technologies are prone to introducing problems when upgrading to the latest and greatest version. Delinea technology has been around for 15 years and unlike common practice with other technologies of waiting to upgrade, Delinea recommends having the latest and greatest versions installed because these will provide greater stability and security. Delinea provides major releases and minor releases every year. The agent is continually receiving security, performance, and feature updates. The administrator tools (consoles, SDKs, APIs) are continually adding functionality that can be pushed to the agent and more support for automation.

Customers should review security updates from Delinea on a periodic basis.

One of the most important things a customer can do is to upgrade the Server Suite Agent once per year to take advantage of the additional functionality/stability offered with latest versions. Server Suite Agents and administrative tools are easy to upgrade, can be done in a modular fashion and are backwards and forwards compatible. The most recent releases can be found at the Downloads section of the Support Portal.

Customers should leverage Enterprise grade deployment framework (supported by technologies like Chef, Puppet, Bladelogic, etc) to automatically deploy the Server Suite Agent and updates. Leverage Chef/Puppet/BladeLogic to automatically deploy and maintain agent configuration parameters and to leverage the Delinea Repo to automatically upgrade target systems in a streamlined fashion. Another option is the Delinea Software Repo for streamlined installation and updates.

Enable NSCD

Nscd is a daemon that provides a cache for the most common name service requests. The default configuration file, /etc/nscd.conf, determines the behavior of the cache daemon. More information on NSCD can be found here.

We recommend enabling nscd on each Server Suite enabled server to maximize the caching performance. The default configuration of nscd will suffice.

Set Group Policies To Govern The Agent Behavior

One of the most powerful features in the Server Suite platform is the ability to centrally push out Group Policy to Linux systems. We recommend deploying at least 1 GPO to your systems so you have the means to centrally configure the agent behavior in your environment. Group policy settings are documented in the *Group Policy Guide*. In the event a change to the Centrify parameters is needed for the environment, a GPO change can quickly deploy the change to the systems.

Set agent parameters

Exclusions of Domains

Server Suite provides robust support for complex active directory environments with varying trust relationships. Many agent parameters can be configured through Group Policy. We often see customers don't cleanup decommissioned domains or have domains in the environment not in scope for Server Suite. We recommend blacklisting the domains that are not in scope or whitelisting only the domains in scope for Server Suite.

An example of excluding, black listing, a domain in /etc/centrifydc/centrifydc.conf is:

adclient.excluded.domains: anvil.acme.com

An example of including, white listing, a domain in /etc/centrifydc.conf is:

adclient.included.domains: anvil.acme.com

Paged Control

To operate the best with the Microsoft Active Directory search optimizer, Server Suite provides a parameter called “adclient.schema.extensions.search.add.paged.control”. We recommend setting this parameter to true to optimize AD lookups.

Suite 2016.1

If the version of the Server Suite Agent for *NIX running is version 5.3.1, part Suite 2016.1, we highly recommend configuring the parameter “adclient.altupn.update.interval: 90000000” in /etc/centrifydc/centrifydc.conf.

These parameters can be deployed via the GPO“Add centrifydc.conf properties” under Computer Configuration > Centrify Settings > DirectControl Settings. See the *Group Policy Guide* for additional information.

Use the Server Suite DB2 Plugin

DB2 systems normally authenticate users against the local Operating System. Therefore, most customers don't think about performance of authentication and lookups when they centralize authentication to Active Directory. However, as customers centralize authentication to Active Directory, performance considerations become more important since DB2 is very user lookup intensive. Customers that leverage DB2 in their environment should consider using the Server Suite DB2 user and group plugin since it delivers enhanced caching to improves the performance of lookups in a DB2 environment that leverages Active Directory for authentication/authorization.

Best Practices for Active Directory Environment

Index the UID Attribute

Many UNIX applications make requests for the uid attribute as part of their inner workings. If the uid attribute is not indexed and applications make frequent requests for uid data, this can have a negative effect on the performance of Domain Controllers. Centrify highly recommends customers index the uid attribute in Active Directory.

Windows Active Directory functional level and Windows Server version

Customers should maintain an upgrade strategy to use a stable and supported Active Directory functional level and the version of Windows server. As of this writing customers should be moving to Windows 2016 functional modes.

Maintain sites and services domain controller topology

A common issue customers come across is Centrify binding to the wrong Domain Controllers. For example, all the users in the US may be authenticating to a domain controller that is not geographically desirable. In most instances, this occurs because the AD Sites and Services definition does not include the subnets of the UNIX/Linux systems or is not updated on a consistent basis.

A process should be defined where the UNIX/Linux networking teams regularly interact with the AD team to assure subnets are added and removed from AD Sites and Services accordingly.

Centrify Access Model Best Practices

Proper definition of global/child zone structure.

A proper Centrify deployment should have a Global Zone with an appropriate number of Child Zones and Computer Roles to drive access across groups of systems. The general recommendation for defining profiles, roles and rights is:

- UNIX enable all users at the Global Zone level
- In addition, UNIX enable users at the child zone level, if attributes need to be different
- for users on the systems in the child zones (ie.different primary group)
- UNIX enable groups at the Child Zone vs. the Global Zone unless the groups need to be visible across all servers
- Always enable ZPA to automate UNIX profile provisioning across all Zones that will have user/group UNIX profiles
- Define Roles and Rights in the Global zone and assign roles at computer roles or zones if appropriate

A common mistake made is the use of too many Child Zones or use Child Zones incorrectly. Limit child zone sprawl. Child zones should be used for specific purposes like:

Segregating systems in different business units

Segregate the management of groups of systems to different administrative groups

Override the UNIX profiles of users and groups across groups of systems.

Another common mistake is managing roles and rights definition throughout the zone hierarchy which makes it difficult to find roles and rights when updates are needed.

Another mistake is using Zones to define access. Instead, use Computer roles to prevent lateral movement, drive an automated access model and to take advantage of performance benefits. Leverage Computer Roles and AD groups to manage system types by likeness of access and create AD user groups in a similar manner. This promotes automation because user access can be granted access/privileges by simply adding users to the right AD groups. Similarly, systems can be provisioned to the right AD group of computers. Computer roles can be defined by application types. For example, “App 1 DEV” App1 PROD”, etc. The goal is to not have to use the access manager UI for provisioning access.

Analyze The Deployment Periodically

As a Server Suite deployment matures, customers should perform a periodic analysis using the Access Manager “Analyze” feature. The analysis highlights problem areas in the Server Suite deployment. For example, the analysis will identify orphaned objects.

Additionally, Centrify recommends periodically reviewing security updates available at our support portal.

Use the Centrify Zone Provisioning Agent

We highly recommend leveraging the Zone Provisioning Agent to automatically provision UNIX profiles for users. Additionally, we recommend two instances of ZPA in large environments. This provides redundancy in the provisioning process. The ZPA service should also be monitored to ensure it is operational.

Deploy Reporting Services and Security Information and Event Management (SIEM)

To maximize the investment, we highly recommend deploying Delinea Reporting Services and SIEM integration. These capabilities provide customers which insight into which users can access which system and security related events the Server Suite Agent reports on. See the following items for more information:

- *Report Administrator's Guide*
- SIEM documentation on the documentation portal
- A Delinea community article

Best Practices for the Audit and Monitoring Service

This section includes the following topics:

Manage the Audit Store Database Size

The Audit Store database needs to be managed according to the company's retention policy which often dictated by the security/compliance team. To assure the audit service performs and scales as required, we recommend keeping the active Audit Store database at most, between 250GB and 500GB in size. Perform a database rotation if your active Audit Store database is larger than 500GB. A database rotation takes the current active database and marks it inactive and makes a new database the active database. See the documentation for how to automate database rotation.

Another approach is to delete audit records and shrink the size of the active database. This approach works well as long as the indexes are also rebuilt. Otherwise, shrinking the database without indexing will lead to fragmented indexes and poor query performance. KB-8472 details how to shrink and re-build the database indexes.

Centrify recommends keeping only databases that are required for auditing purposes attached to the audit infrastructure. The databases that are not needed should be detached. Customers often forget to detach the databases that are outside the company's normal live data/retention policies. Too many attached Audit Store databases result into poor query performance and increased load on the Management database. Periodically review the list of attached Audit Store databases and detach the ones that are no longer needed to be online as per the retention policy.

Maintain the audit store database index

It is recommended to maintain the audit store database's indexes regularly. This can be done by setting up a simple SQL job to reorganize the indexes if they are 5%-30% fragmented and rebuild the indexes if they're more than 30% fragmented. KB-8472 details how shrink and re-build the database indexes.

Configure SQL Server

There are several SQL specific configurations and server settings that can affect performance and operation.

Avoid deploying the Audit Store databases in a SQL availability group unless it's required by the company's compliance policies.

Configure SQL Server power settings to be set to Balanced instead of High Performance.

SQL Server has a setting called Max Server Memory that controls the maximum amount of physical memor that can be consumed by the SQL Server's buffer pool. An incorrectly configured Max Server Memory may either result into

the SQL engine causing high IO or OS/other programs starving for more memory. Refer to the “configuring the maximum memory for audit store databases” section of the Auditing Administrator's Guide and always configure this value as recommended before the deployment begins.

If you're expecting a database server to get migrated/retired in the near future, it's better to create a CNAME alias in DNS for the current database server and specify the alias everywhere (e.g. when creating a new Management database) rather than specifying the actual host name. This will prevent scenarios where the database server is not found after a migration.

Audit and Monitoring Architecture

The audit architecture includes several components to ensure a smooth operating and secure audit environment. A Collector is the service that collects audit records from servers being audited and stores them in the audit store database. Avoid deploying the collector on the same machine as the active Audit Store database's SQL Server.

When using the Server Suite Agent for Windows to audit sessions, configure data capture at native color depth when auditing systems with many concurrent users (such as Citrix XenApp server). When not capturing at native color depth, the DirectAudit daemon has to transform the captured data to its target format which ends up consuming CPU cycles. To automatically set native color depth at installation time, see the documentation.

Grant Audit Installation Rights To Administrator Groups

Centrify Audit Administrators have specific privileges to manage and configure the Server Suite audit configuration. A common mistake is rights are assigned to specific AD accounts vs. AD groups or rights are not delegated at all. When employees leave the company and those AD accounts are disabled, the Audit services becomes inaccessible. To avoid this, Centrify recommends rights over the Audit installation are delegated to AD groups and administrators/auditors are placed into the proper AD groups. This will ensure that all administrators within that security group will have access to configure/modify the audit settings in the event that a specific employee leaves the organization.

Delinea Relationship Best Practices

Monthly Cadence Call with Delinea

Customers can schedule a monthly cadence call with Delinea Account Team, Support, and Engineering to ensure that best practices and customer requirements are consistently being communicated.

Customers should have the contact information of their account team (Account executive, customer success manager, and systems engineer). This account team can help escalate requests internally within Delinea, handle licensing questions and feature requests.

Get Your Annual Delinea Healthcheck

Customers are entitled to annual healthchecks provided by the Delinea account team. Delinea conducts healthchecks in varying scope. A basic healthcheck can be conducted by a Delinea Systems Engineer in a few hours and can provide insight into the usage of Centrify, risks and areas for improvement. Additionally, more advanced healthchecks can be provided in a 3 to 5 day paid engagement with Delinea professional services to delve deep into the environment to provide recommendations for security and operational improvement as well as address identified issues.

Attend Annual Delinea Update Meetings

Customers are entitled to annual update meetings with the Delinea Account and Product Management to understand new feature availability and to submit/track enhancements. Customers often find that Delinea has provided additional capability in a new release that addresses new requirements they are entitled to. Additionally, this helps customers to understand the product roadmap and direction for Delinea and promotes a partnership between Delinea and the customer.