One Identity Authentication Services 4.1.8

Upgrade Guide

Legend

WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

IMPORTANT, NOTE, TIP, MOBILE, or VIDEO: An information icon indicates supporting information.

Authentication Services Upgrade Guide
Updated - April 2019
Version - 4.1.8

Contents

Privileged Access Suite for Unix

Unix security simplified

Privileged Access Suite for Unix solves the inherent security and administration issues of Unix-based systems (including Linux and Mac OS X) while making satisfying compliance requirements easier. It unifies and consolidates identities, assigns individual accountability, and enables centralized reporting for user and administrator access to Unix. The Privileged Access Suite for Unix combines an Active Directory bridge and root delegation solutions under a unified console that grants organizations centralized visibility and streamlined administration of identities and access rights across their entire Unix environment.

Active Directory bridge

Achieve unified access control, authentication, authorization, and identity administration for Unix, Linux, and Mac OS X systems by extending them into Active Directory (AD) and taking advantage of AD's inherent benefits. Patented technology allows non-Windows resources to become part of the AD trusted realm, and extends AD's security, compliance, and Kerberos-based authentication capabilities to Unix, Linux, and Mac OS X. See www.oneidentity.com/products/authentication-services/ for more information about the Active Directory Bridge product.

Root delegation

The Privileged Access Suite for Unix offers two different approaches to delegating the Unix root account. The suite either *enhances* or *replaces* sudo, depending on your needs.

- By choosing to enhance sudo, you will keep everything you know and love about sudo while enhancing it with features like a central sudo policy server, centralized keystroke logs, a sudo event log, and compliance reports for who can do what with sudo.

  See www.oneidentity.com/products/privilege-manager-for-sudo/ for more information about enhancing sudo.
- By choosing to replace sudo, you will still be able to delegate the Unix root privilege based on centralized policy reporting on access rights, but with a more granular permission and the ability to log keystrokes on all activities from the time a user logs

in, not just the commands that are prefixed with "sudo." In addition, this option implements several additional security features like restricted shells, remote host command execution, and hardened binaries that remove the ability to escape out of commands and gain undetected elevated access.

See www.oneidentity.com/products/privilege-manager-for-unix/ for more information about replacing sudo.

Privileged Access Suite for Unix

Privileged Access Suite for Unix offers two editions: *Standard* edition and *Advanced* edition. Both editions include the Management Console for Unix, a common mangement console that provides a consolidated view and centralized point of management for local Unix users and groups; and Authentication Services, patented technology that allows organizations to extend the security and compliance of Active Directory to Unix, Linux, and Mac OS X platforms and enterprise applications. In addition:

- The *Standard* edition licenses you for Privilege Manager for Sudo.
- The *Advanced* edition licenses you for Privilege Manager for Unix.

One Identity recommends that you follow these steps:

1. Install Authentication Services on one machine, so you can set up your Active Directory Forest.
2. Install Management Console for Unix, so you can perform all the other installation steps from the mangement console.
3. Add and profile hosts using the mangement console.
4. Configure the console to use Active Directory.
5. Deploy client software to remote hosts.

   Depending on which Privileged Access Suite for Unix edition you have purchased, deploy one of the following:

   - **Privilege Manager for Unix** software (that is, Privilege Manager Agent packages)

     -OR-

   - **Privilege Manager for Sudo** software (that is, Sudo Plugin packages)

About this guide

The *Authentication Services Upgrade Guide* is intended for Windows, Unix*, Linux, and Macintosh system administrators, network administrators, consultants, analysts, and any other IT professionals who will be upgrading Authentication Services to version 4.1 from any previous release. This guide walks you through one simple approach to upgrading Authentication Services, highlighting the changes and enhancements associated with installing and configuring Authentication Services using Management Console for Unix.

Of course, you can upgrade and install Authentication Services without using Management Console for Unix. You can find those instructions in the *Authentication Services Installation Guide*.

NOTE: Authentication Services versions 3.*x* and 4.*x* can both run in the same domain (on different machines).

These are the basic Authentication Services upgrade steps:

1. Upgrade from 3.5 to 4.1 considerations on page 27
2. Upgrade the web console on page 36
3. Upgrade Authentication Services Windows components on page 42
4. Configure Active Directory for Authentication Services on page 44
5. Configure Unix agent components on page 50

* The term "Unix" is used informally throughout the Authentication Services documentation to denote any operating system that closely resembles the trademarked system, UNIX.

2

Introducing One Identity Authentication Services

One Identity Authentication Services is patented technology that enables organizations to extend the security and compliance of Active Directory to Unix, Linux, and Mac OS X platforms and enterprise applications. It addresses the compliance need for cross-platform access control, the operational need for centralized authentication and single sign-on, and enables the unification of identities and directories for simplified identity and access management.

Upgrade requirements

You can upgrade Authentication Services from any existing supported version of the product by installing Authentication Services on the computer where the old version was installed.

To upgrade Authentication Services, you must have local administrator rights to:

- create a container and a child container in Active Directory
- join a Unix host to the Active Directory domain

NOTE: Have your license available for the Setup wizard.

Licensing Authentication Services

Authentication Services must be licensed in order for Active Directory users to authenticate on Unix and Mac OS X hosts.

NOTE: When upgrading, Authentication Services continues to use licenses from previous versions. This allows the upgrade to take place without having to distribute new license files first. Any VAS 3.*x* or higher license is valid for Authentication Services 4.1.

NOTE: While you can install and configure Authentication Services on Windows and use the included management tools to Unix-enable users and groups in Active Directory without installing a license, you must have the Authentication Services license installed for full functionality.

Contact your account representative for a license.

System requirements

Prior to installing Authentication Services, ensure your system meets the minimum hardware and software requirements for your platform. Authentication Services consists of Windows management tools and Unix client agent components.

Windows management tools requirements

The following are the minimum requirements for installing Authentication Services in your Windows environment.

Table 1: Authentication Services Windows requirements

System requirements	
Supported Windows Platforms	You can install Authentication Services on 32-bit or 64-bit editions of the following configurations: • Windows XP SP2 (or later) • Windows Vista • Windows 7 • Windows 8 • Windows Server 2003 SP1 (or later) • Windows Server 2008 • Windows Server 2008 R2 • Windows Server 2012 • Windows Server 2012 R2 • Windows Server 2016 • Windows Server 2019

System requirements	
	NOTE: Due to tightened security, when running Authentication Services Control Center on Windows 2008 R2 (or higher) operating system, functioning as a domain controller, the process must be elevated or you must add authenticated users to the Distributed COM Users group on the computer. As a best practice, One Identity does not recommend that you install or run the Authentication Services Windows components on Active Directory domain controllers. The recommended configuration is to install the Authentication Services Windows components on an administrative workstation.
Prerequisite Windows Software	You can download all of the following prerequisite software free from the Microsoft website: • Windows Installer 3.1 (http://support.microsoft.com/kb/893803) • Microsoft .NET Framework 3.5 SP1 or higher • Windows PowerShell 1.0 or higher (http://support.microsoft.com/kb/968929) If any of the prerequisites are missing, the Authentication Services installer suspends the installation process to allow you to download the required component; it then continues the install.

Authentication Services Windows components

Authentication Services includes the following Windows components.

Table 2: Windows components

Windows component	Description
Authentication Services Control Center	A single console for access to all of the tools and configuration settings for Authentication Services.
Active Directory Users and Computers MMC Snapin Extensions	Unix management extensions for Active Directory users and groups.
Group Policy Management Editor MMC Snapin Extensions	Group Policy extensions for management of Unix, Linux, and Mac OS X.
RFC2307 NIS Map Editor MMC Snapin	Provides the ability to manage NIS data in Active Directory.
NIS Map Import Wizard	Imports NIS data into Active Directory.
Unix Account Import Wizard	Imports Unix identity data into Active Directory.
Authentication Services Power-	Provides the ability to script Unix management tasks.

Windows component	Description
Shell cmdlets	
Documentation	Full product documentation and online help.

NOTE: The VAS Configuration Utility is no longer included. Instead the Control Center provides access to all preferences and tools. If you were using the custom schema functionality of the VAS Configuration Utility, be sure to configure the same settings in the Control Center under **Preferences | Custom Unix Attributes**.

Any previous version of the Authentication Services Windows components are automatically uninstalled before the Authentication Services 4.1 install proceeds.

Windows permissions

To install Authentication Services on Windows, you must have:

- Local administrator rights
- Rights to create and delete all child objects in the container where you will install the configuration settings (first-time only)

Authenticated Users must have rights to read `cn`, `displayName`, `description`, and `whenCreated` attributes for container objects in the application configuration location. To change Active Directory configuration settings, Administrators must have rights to Create Child Object (container) and Write Attribute for `cn`, `displayName`, `description`, and `showInAdvancedViewOnly` in the application configuration location.

Table 3: Required Windows permissions

Rights required	For user	Object class	Attributes
Create Child Object	Authentication Services Administrators Only	Container	
Delete Child Object	Authentication Services Administrators Only	Container	
Delete Child Object	Authentication Services Administrators Only	Container	
Write Attribute	Authentication Services Administrators Only	Container	`cn, displayName, description, showInAdvancedViewOnly`
Read Attribute	Authenticated Users	Container	`cn, displayName, description, whenCreated`

Unix agent requirements

NOTE: To install Authentication Services on Unix, Linux, or Mac OS X, you must have root access rights.

Click www.oneidentity.com/products/authentication-services/ to view a list of supported Unix and Linux platforms for Authentication Services 4.1.

With Authentication Services 4.1, Linux platforms require glibc 2.4 or greater.

For maximum security and performance, before you begin the installation, make sure that you have the latest patches for your operating system version.

Table 4: Unix agent: Patch level requirements

Platform	Patch level
Solaris 8 SPARC	108993-55 or greater
Solaris 8 X86	108994-01 or greater 112757-01 or greater
Solaris 9 SPARC	112874-37 or greater 112960-14 or greater 113319-22 or greater
Solaris 9 X86	114432-37 or greater
Solaris 10 SPARC	127127-11 or greater
Solaris 10 x86	127128-11 or greater
AIX 5.3	OS level 5300-05 or greater
AIX 6.1	OS level 5300-05 or greater
AIX 7.1	OS level 5300-05 or greater
HPUX 11.11	GOLDQPK11i - GOLDBASE11i GOLDAPPS11i quality packs BUNDLE11i - Patch bundle linker tools cumulative patch (PHSS_30970 or greater)
HPUX 11.23	MAINTPACK E0306 or greater

NOTE: One Identity recommends that you run the Preflight utility to check for supported operating system and correct operating system patches.

For more information, see *Running Preflight* in the *Authentication Services Installation Guide*.

Authentication Services Unix components

Authentication Services includes the following Unix components.

Table 5: Authentication Services Unix components

Unix component	Description
vasd	The Authentication Services agent background process that manages the persistent cache of Active Directory information used by the other Authentication Services components. `vasd` is installed as a system service. You can start and stop `vasd` using the standard service start/stop mechanism for your platform. `vasd` is installed by the **vasclnt** package.
vastool	The Authentication Services command line administration utility that allows you to join a Unix host to an Active Directory Domain; access and modify information about users, groups, and computers in Active Directory; and configure the Authentication Services components. `vastool` is installed at `/opt/quest/bin/vastool`. `vastool` is installed by the **vasclnt** package.
vgptool	A command line utility that allows you to manage the application of Group Policy settings to Authentication Services clients. `vgptool` is installed at `/opt/quest/bin/vgptool`. `vgptool` is installed by the **vasgp** package.
oat (Ownership Alignment Tool)	A command line utility that allows you to modify file ownership on local Unix hosts to match user accounts in Active Directory. `oat` is installed at `/opt/quest/libexec/oat/oat`. `oat` is installed by the **vasclnt** package.
LDAP proxy	A background process that secures the authentication channel for applications using LDAP bind to authenticate users without introducing the overhead of configuring secure LDAP (LDAPS). The LDAP proxy is installed by the **vasproxy** package.
NIS proxy	A background process that acts as a NIS server which can provide backwards compatibility with existing NIS infrastructure. The NIS proxy is installed by the **vasyp** package.
SDK package	The **vasdev** package, the Authentication Services programming API.

Authentication Services permissions matrix

The following table details the permissions required for full Authentication Services functionality.

Table 6: Authentication Services: Required permissions

Function	Active Directory permissions	Local client permissions
Authentication Services Application Configuration: creation	Location in Active Directory with Create Container Object rights	N/A
Authentication Services Application Configuration: changes • Unix Global Settings • Licensing • Custom Unix Attributes	Update permission to the containers created above (no particular permissions if you are the one who created it)	N/A
Schema optimization	Schema Administrator rights	N/A
Display Specifier Registration	Enterprise Administrator rights	N/A
Editing Users	Administrator rights	N/A
Create any group policy objects	Group Policy Creator Owners rights	N/A
RFC 2307 NIS Import Map Wizard	Location in Active Directory with Create Container Object rights (you create containers for each NIS map)	N/A
Unix Account Import Wizard	Administrator rights (you are creating new accounts)	N/A
Logging Options	Write permissions to the file system folder where you want to create the logs	N/A
vasd daemon	The client computer object is expected to have read access to user and group attributes, which is the default. In order for Authentication Services to update the host object operating system attributes automatically, set the following rights for "SELF" on the client computer object: **Write Operating System**, **Write operatingSystemHotfix**, and **Write**	vasd must run as root

Function	Active Directory permissions	Local client permissions
	operatingSystemServicePack.	
QAS/VAS PAM module	N/A (updated by means of `vasd`)	Any local user
QAS/VAS NSS module `vastool nss`	N/A (updated by means of `vasd`)	Any local user
`vastool` command-line tool	Depends on which `vastool` command is run	Any local user for most commands
`vastool join` `vastool unjoin`	Computer creation or deletion permissions in the desired container	root
`vastool configure` `vastool unconfigure`	N/A	root
`vastool search` `vastool attrs`	Read permission for the desired objects (regular Active Directory user)	Any local user
`vastool setattrs`	Write permissions for the desired object	Any local user
`vastool cache`	N/A	Run as root if you want all tables including authcache
`vastool create`	Permissions to create new users, groups, and computers as specified	Any local user; root needed to create a new local computer
`vastool delete`	Permissions to delete existing users, groups, or computers as specified; permissions to remove the keytab entry for the host object created (root or write permissions in the directory and the file)	Any local user
`vastool flush`	The client computer object is expected to have read access to user and group attributes, which should be the default	root

Function	Active Directory permissions	Local client permissions
`vastool group add` `vastool group del`	Permission to modify group membership	Any local user
`vastool group hasmember`	Read permission for the desired objects (regular Active Directory user)	Any local user
`vastool info { site | domain | domain -n | forest-root | forest-root -dn | server | acl }`	N/A	Any local user
`vastool info { id | domains | domains -dn | adsecurity | toconf }`	Read permission for the desired objects (regular Active Directory user)	Any local user

`vastool isvas`
`vastool inspect`
`vastool license`
	N/A	Any local user

`vastool kinit`
`vastool klist`
`vastool kdestroy`
	Local client needs permissions to modify the keytab specified; default is the computer object, which is root.	Any local user

`vastool ktutil`	N/A	root if you are using the default `host.keytab` file
`vastool list` (with `-l` option)	Read permission for the desired objects (regular Active Directory user)	Any local user
`vastool load`	Permissions to create users and groups in the desired container	Any local user
`vastool merge` `vastool unmerge`	N/A	root
`vastool passwd`	Regular Active Directory user	Any local user

Function	Active Directory permissions	Local client permissions
`vastool passwd <AD user>`	Active Directory user with password reset permission	Any local user
`vastool schema list` `vastool schema detect`	Regular Active Directory user	Any local user
`vastool schema cache`	Regular Active Directory user	root (to modify the local cache file)
`vastool service list`	Regular Active Directory user	Any local user
`vastool service { create | delete }`	Active Directory user with permission to create/delete service principals in desired container	N/A
`vastool smartcard`	N/A	root
`vastool status`	N/A	root
`vastool timesync`	N/A	root, if you only query the time from AD, you can run as any local user
`vastool user { enable | disable }`	Modify permissions on the AD Object	Any local user
`vastool user { checkaccess | checkconflict }`	N/A	Any local user
`vastool user checklogin`	Access to Active Directory users password	Any local user

Authentication Services encryption types

The following table details the encryption types used in Authentication Services.

Table 7: Authentication Services: Encryption types

Encryption types	Specification	Active Directory version	Authentication Services version
KERB_ENCTYPE_DES_CBC_CRC			
CRC32	RFC 3961	All	All
KERB_ENCTYPE_DES_CBC_MD5			
RSA-MD5	RFC 3961	All	All
KERB_ENCTYPE_RC4_HMAC_MD5			
RC4-HMAC-MD5	RFC 4757	All	All
KERB_ENCTYPE_AES128_CTS_HMAC_SHA1_96			
HMAC-SHA1-96-AES128	RFC 3961	Windows Server 2008 +	3.3.2+
KERB_ENCTYPE_AES256_CTS_HMAC_SHA1_96			
HMAC-SHA1-96-AES256	RFC 3961	Windows Server 2008 +	3.3.2+

Management Console for Unix requirements

One Identity recommends that you install One Identity Management Console for Unix, a separate One Identity product that provides a mangement console that is a powerful and easy-to-use tool that dramatically simplifies deployment of Authentication Services agents to your clients. The mangement console streamlines the overall management of your Unix, Linux, and Mac OS X hosts by enabling centralized management of local Unix users and groups and providing granular reports on key data and attributes.

Prior to installing Management Console for Unix, ensure your system meets the minimum hardware and software requirements for your platform.

Table 8: Management Console for Unix: Hardware and software requirements

Component	Requirements
Supported platforms	Can be installed on the following configurations: • Windows x86 (32-bit) • Windows x86-64 (64-bit) • Unix/Linux systems for which Java 8 is available
Server require-ments	The Management Console for Unix server requires Java 8 (also referred to as JRE 8, JDK 8, JRE 1.8, and JDK 1.8).
Managed Host	Click www.oneidentity.com/products/authentication-services/ to view

Component	Requirements
Requirements	a list of Unix, Linux, and Mac platforms that support Authentication Services. Click www.oneidentity.com/products/privilege-manager-for-unix/ to review a list of Unix and Linux platforms that support Privilege Manager for Unix. Click www.oneidentity.com/products/privilege-manager-for-sudo/ to review a list of Unix, Linux, and Mac platforms that support Privilege Manager for Sudo. NOTE: To enable the Management Console for Unix server to interact with the host, you must install both an SSH server (that is, sshd) and an SSH client on each managed host. Both OpenSSH 2.5 (and higher) and Tectia SSH 5.0 (and higher) are supported. NOTE: Management Console for Unix does not support Security-Enhanced Linux (SELinux) NOTE: When you install Authentication Services on Solaris 10 (SPARC - 32/64-bit), the Solaris 10 packages are installed.
Default memory requirement	1024 MB NOTE: See *JVM memory tuning suggestions* in the *One Identity Management Console for Unix Administration Guide* for information about changing the default memory allocation setting in the configuration file.

Network requirements

Authentication Services must be able to communicate with Active Directory, including domain controllers, global catalogs, and DNS servers using Kerberos, LDAP, and DNS protocols. The following table summarizes the network ports that must be open and their function.

Table 9: Network ports

Port	Function
389	Used for LDAP searches against Active Directory Domain Controllers. TCP is normally used, but UDP is used when detecting Active Directory site membership.
3268	Used for LDAP searches against Active Directory Global Catalogs. TCP is always used when searching against the Global Catalog.

Port	Function
88	Used for Kerberos authentication and Kerberos service ticket requests against Active Directory Domain Controllers. TCP is used by default.
464	Used for changing and setting passwords against Active Directory using the Kerberos change password protocol. Authentication Services always uses TCP for password operations.
53	Used for DNS. Since Authentication Services uses DNS to locate domain controllers, DNS servers used by the Unix hosts must serve Active Directory DNS SRV records. Both UDP and TCP are used.
123	UDP only. Used for time-synchronization with Active Directory.
445	CIFS port used to enable the client to retrieve configured group policy.

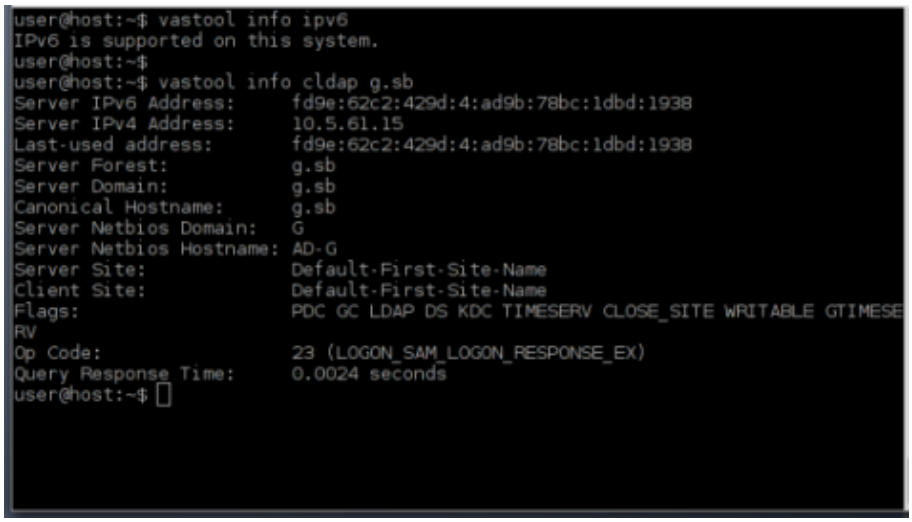
NOTE: Authentication Services, by default, operates as a client, initiating connections. It does not require any firewall exceptions for incoming traffic.

3

What's new in Authentication Services 4.1

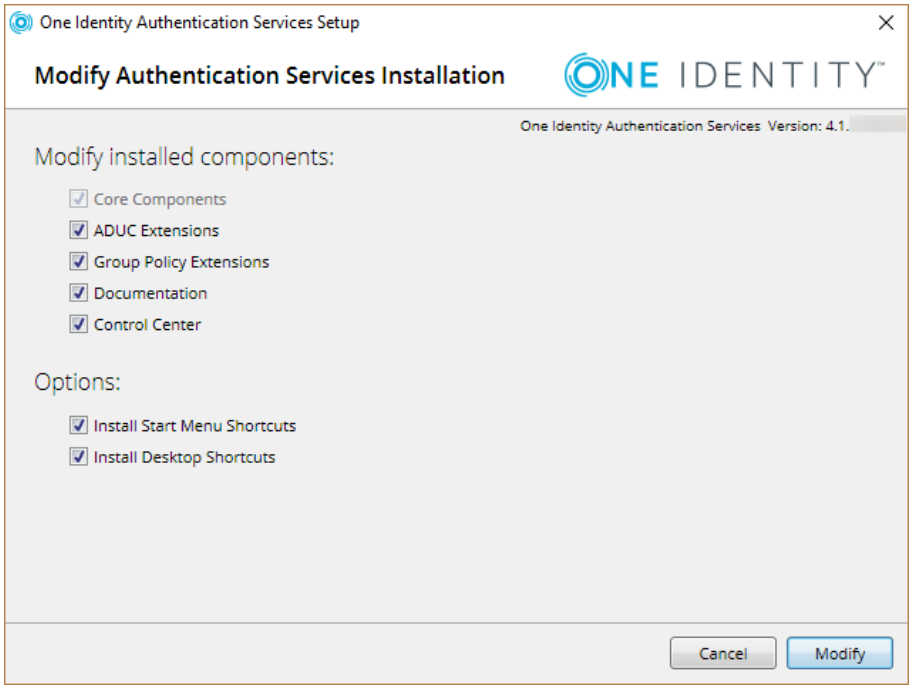
Authentication Services, the solution that pioneered the "Active Directory Bridge" market, continues to lead the way with powerful and innovative new capabilities that make heterogeneous identity and access management even more efficient, secure, and compliant. Authentication Services 4.1 features include:

- **Upgrade Without Reboot** – This version of Authentication Services adds the functionality required so that future upgrades will no longer require a system reboot. Some customer deployments of Authentication Services have been running on old versions for long periods of time because of the difficulties of scheduling server down time. With Authentication Services 4.1 deployed as the foundation, future releases will allow customers to deploy upgrades without impacting running services or rebooting.
- **IPv6 Support** – Authentication Services now supports hosts running in full IPv6 environments. Authentication Services automatically uses IPv6 when it is available; it uses IPv4 when IPv6 is not available or significantly slower than IPv4. IPv6 is available in Authentication Services on most recent operating systems, but is operating system dependent. Run `vastool info ipv6` to determine whether IPv6 is available on each client. Authentication Services operates in IPv4-only, IPv6-only, or dual-stack environments; no special configuration is required. Active Directory servers must be running Windows 2008 or later for IPv6 communication.

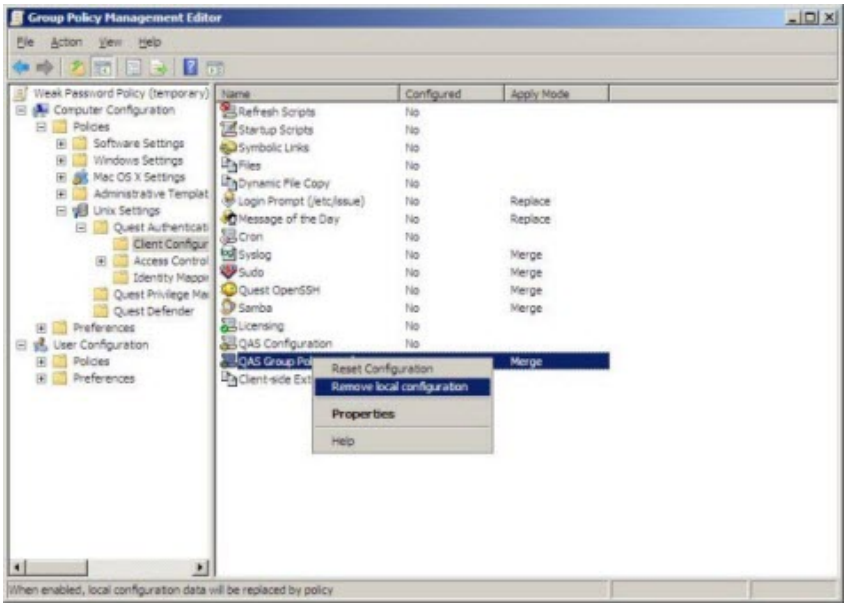


NOTE: Authentication Services uses IPv6 when the operating system's DNS resolver correctly supports mapping of IPv4 addresses to IPv6 addresses. If a problem with address mapping is detected, Authentication Services operates in IPv4-only mode, even if an IPv6 address is assigned and other applications use IPv6.

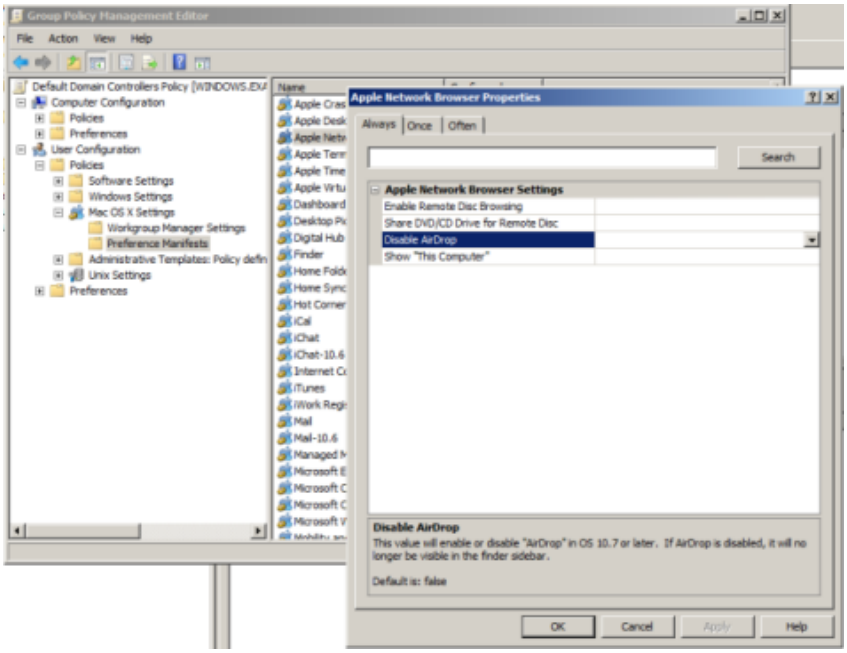
- **Customizable Windows Components Installer** - The Windows installer was upgraded to be fully customizable so that you can install individual components. For example, you can install an individual MMC snap-in without installing the entire Control Center application.



- **Authentication Services Group Policy Updates**:
  - Support for the native Active Directory 'Apply' right.
  - Ability to specify "merging" or "replacing" several local file settings in the GPO. For example, you can configure `users.allow` to be delivered to every system with the contents overwriting any changes made to the local copy of `users.allow`.



  - A new 'NetWork Browser' preference manifest setting for MAC Group Policy that allows you to deactivate AirDrop.



NOTE: When upgrading Authentication Services, you must manually add this new preference manifest. Refer to the *Preference Manifest Settings* topic in the *Authentication Services Mac OS X/macOS Administration Guide* for the procedure *To Add a Preference Manifest*.

- **Group Policy for Certificate autoenrollment** - Certificate Autoenrollment provides a quick and simple way to issue and renew certificates for Mac OS X users and systems from Windows 2008 R2 Certificate Enrollment Web Services. In this release you can configure Certificate autoenrollment with Group Policy. Certificate autoenrollment includes the ability to:
  - Automatically enroll X509 Certificates based on Microsoft Certificate Enrollment Policy
  - Renew certificates that are close to expiration according to policy
  - Automatically install newly enrolled Certificates into the Mac OS X Keychain
  - Support both user and machine certificate policy

  NOTE: Group Policy for Certificate autoenrollment is not supported in the Pre-Release Evaluation Guide software.
- **Management Console for Unix 2.5 Updates**:
  - Ability to manage access control settings (`users.allow`)
  - Ability to managePrivilege Manager for Unix (sold separately)

New and deprecated Unix platform support

Authentication Services 4.1 supports Mac OS X 10.8 and later. Support has been dropped for Mac OS X 10.7 and earlier.

For the most accurate list of supported platforms, please consult the Authentication Services Platform Support table on *Authentication Services Platform Support*.

4

Upgrade from 3.5 to 4.1 considerations

There were some significant changes in Authentication Services 4.0. Some of the changes could result in unexpected behavior unless you take the appropriate action before upgrading.

Active Directory settings changes

In VAS 3.5 settings that affected the Active Directory Users and Computers MMC snap-in behavior were set in the VAS Configuration Utility and only affected the local workstation. Authentication Services 4.*x* no longer includes the VAS Configuration Utility and has moved the Active Directory Users and Computers MMC snap-in behavior settings to the Control Center in Active Directory. Because the settings are stored in Active Directory, they affect the behavior of all workstations running Authentication Services 4.*x* in the mangement console, ADUC snap-ins and PowerShell.

To verify these settings

1. From the Control Center navigate to the **Preferences** view.
2. Validate the **Global Unix Options** and the **Custom Unix Attributes**.

UID and GID changes

To help you avoid ID conflicts with existing local users, in Authentication Services 4.*x* you can set global minimum and maximum values for UID number and GID number in the Control Center on the *Preferences* page under Global Unix Options. Authentication Services management tools enforce these minimum and maximum values.

NOTE: Authentication Services 4.*x* accepts existing UID and GID numbers, however if you modify them later, you must conform to the global minimum and maximum values.

By default, Authentication Services 4.*x* uses a new algorithm for generating unique Unix ID numbers. Unix ID numbers are generated based on the object GUID of the Active Directory user or group. You can modify this behavior in Control Center.

The following three algorithms are supported:

- Object GUID Hash:

  The ID is based on a hash of the object GUID.

- Samba:

  The ID is based on a combination of the SID and object RID.

- Legacy:

  The ID is generated by searching Active Directory for existing IDs. This is the algorithm used in 3.*x*.

If the Object GUID Hash or Samba methods do not produce a unique ID, the Legacy algorithm is used as a fallback to produce a unique ID.

User identity specification changes

Authentication Services 4.*x* uses new user name formats for identifying users and groups in configuration files.

Authentication Services daemon changes for upgrade

vasd Caching and vasgpd Group Policy Daemons

NOTE: The changes made in Authentication Services 4.*x* may affect any monitoring scripts that you created for watching the `vasd` or `vasgpd` daemons.

In VAS 3.*x* `vasd` and `vasgpd` (Group Policy update daemon) were separate processes delivered in separate packages. In Authentication Services 4.*x*, the functionality of `vasgpd` has been absorbed into `vasd`, eliminating `vasgpd`.

However, please note:

- you must still install the `vasgp` package in order to utilize Group Policy on the Unix host; and
- `vastool` no longer stops `vasd` during a flush operation to allow the daemon to supply Name Service data.

vasd Changes

To improve the stability and integrity of the local identity cache in Authentication Services 4.*x*, One Identity updated `vasd` to provide better isolation of the processes responsible for accessing the local identity cache.

In a typical 3.*x* environment, `vasd` was split into a parent process, with a single child process, whose sole responsibility was to maintain the local cache and respond to all update requests from the Name Service and Authentication modules.

Authentication Services 4.*x* changed the process hierarchy and now uses five separate but related `vasd` processes which allow `vasd` to ensure cache integrity, as well as maintain responsiveness from all requests. It also removes the need to start additional processes to handle legacy password hash and netgroup data requests.

One Identity designed Authentication Services 4.1 to be backwards compatible. There are no configuration changes you need to make to take advantage of this improvement.

Authentication Services configuration file changes

Authentication Services 4.*x* has extended the syntax of many of the host configuration files to allow you to specify users and groups by the more commonly used `DOMAIN\sAMAccountName` identifier.

The following configuration files are affected:

- Account overrides

  ```
  /etc/opt/quest/vas/user-override
  /etc/opt/quest/vas/group-override
  ```

- Access control

  ```
  /etc/opt/quest/vas/users.allow
  /etc/opt/quest/vas/users.deny
  ```

- Client configuration

  ```
  /etc/opt/quest/vas/vas.conf
  ```

The extended syntax does not affect configuration entries that were configured and working under previous versions of Authentication Services. The new syntax provides an optional format that you can use in the future. Group Policy settings use the new format if configured with the Group Policy object editor.

Account overrides

User Account Overrides

Entries in the `user-override` file have the form:

```
<identifier>:<Unix name>:<uid>:<primary gid>:<gecos>:<home directory>:<login shell>
```

Table 10: User Account Override identifiers

Identifier	Description
localuser@example.com	For backwards compatibility with previous versions of Authentication Services, any identifier in the file that contains an @ character is interpreted as the LDAP userPrincipalName of an Active Directory user.
localgroup	For backwards compatibility with previous versions of Authentication Services, any simple name in the file is interpreted as the name of an Active Directory group.
EXAMPLE\localuser or EXAMPLE\localgroup	In previous versions of Authentication Services, the agent assumed that this identifier was only used for Active Directory groups. In Authentication Services, any identifier that contains an '\' character is interpreted as the `DOMAIN\sAMAccountName` of an Active Directory object. That object may be either a user or a group.

Group Account Overrides

Entries in the `group-override` file have the form:

```
<identifier>:<Unix name>:<gid>:<member list>
```

Table 11: Group Account Override identifiers

Identifier	Description
localgroup	For backwards compatibility with previous versions of Authentication Services, any simple name in the file is interpreted as the name of an Active Directory group of the joined domain.
EXAMPLE\localgroup	In Authentication Services, any identifier that contains an '\' character is interpreted as the `DOMAIN\sAMAccountName` of an Active Directory object. In this file, that object is always a group.

One Identity designed Authentication Services 4.1 to be backwards compatible. There are no configuration changes you need to make to take advantage of this improvement.

Access control changes

The `users.allow` and `users.deny` files contain a list of identifiers, one per line.

Table 12: Account Control identifiers

Identifiers	Description
localuser@example.com	For backwards compatibility with previous versions of Authentication Services, any identifier in the file that contains an @ character is interpreted as the LDAP userPrincipalName of an Active Directory user.
localgroup	For backwards compatibility with previous versions of Authentication Services, any simple name in the file is interpreted as the name of an Active Directory group.
EXAMPLE\localuser EXAMPLE\localgroup	In previous versions of Authentication Services, the agent assumed that this identifier was only used for Active Directory groups. In Authentication Services 4.*x*, any identifier that contains an '\' character is interpreted as the `DOMAIN\sAMAccountName` of an Active Directory object. That object may be either a user or a group.
@example.com	Any identifier that begins with @ indicates a domain. This allows you to specify all users in the domain.
ou=foo,dc=example,dc=com	Any identifier in DN format specifies a container or OU. This allows you to specify all users under the container or OU.

One Identity designed Authentication Services 4.1 to be backwards compatible. There are no configuration changes you need to make to take advantage of this improvement.

Changes in access control with service-level files

In VAS 3.*x*, if either the `<service>.allow` or `<service>.deny` service-level access control file was missing, then the corresponding `users.allow` or `users.deny` file would be used.

In Authentication Services 4.*x*, any missing service-level access control file is treated as an empty file and thus treated as though there were no corresponding *allow* or *deny* rules for that service.

Client configuration changes

The `vas.conf` configuration file has four settings where you can specify a user, a group, or a list of users or groups. Authentication Services 4.0 modified these settings to allow you to

use the DOMAIN\sAMAccountName identifier to list any Active Directory user or group.

The following settings are affected.

Table 13: Client Configuration Changes

Section	Key	Notes
vas_ macos	admin-users	A comma-separated list of users, groups, or both. An identifier with an @ character is interpreted as the LDAP userPrincipalName of an Active Directory user. An identifier with an '\' character is interpreted as the DOMAIN\sAMAccountName of an Active Directory user or group. Simple names are not allowed.
vas_ auth	mapped-root-user	Only a user may be specified. An identifier with an @ character is interpreted as the LDAP userPrincipalName. An identifier with an '\' character is interpreted as the DOMAIN\sAMAccountName of an Active Directory user or group. Simple names are not allowed.
vasd	perm-disconnected-users	A comma-separated list of users, groups, or both. An identifier with an @ character is interpreted as the LDAP userPrincipalName of an Active Directory user. An identifier with an '\' character is interpreted as the DOMAIN\sAMAccountName of an Active Directory user. Simple names are interpreted as the sAMaccountName of an Active Directory group.
vasd	workstation-mode-users-preload	A comma-separated list of groups. An identifier with an '\' character is interpreted as the DOMAIN\sAMAccountName of an Active Directory group. Simple names are interpreted as the sAMaccountName of an Active Directory group of the joined domain.

One Identity designed Authentication Services 4.1 to be backwards compatible.

vas.conf [nss_vas] option changes

Authentication Services 4.*x* changed the default for the root-update-mode option. In VAS 3.5 the default option was force. In Authentication Services 4.*x*; the default is force-if-missing. This causes the nss_vas module to force an update to the vasd cache whenever a process running as root performs a name search for a user that is not already in the identity cache.

Schema configuration changes

In VAS 3.5.*x* all schema configuration was stored on each host machine as local settings in the agent configuration file (`vas.conf`). Because of this, you had to modify schema configuration on a client-by-client basis. In Authentication Services 4.*x*, the majority of these schema settings are stored globally in the Active Directory configuration. This results in the deprecation of a number of client-specific schema customization options, including:

- groupname-attr-name
- uid-number-attr-name
- gid-number-attr-name
- gecos-attr-name
- homedir-attr-name
- login-shell-attr-name

If you are using any of these settings in an existing 3.*x* install, you need to ensure that Active Directory has been configured with the correct schema mapping information before proceeding with agent upgrade.

Additionally in Authentication Services 4.*x*, the agent no longer uses the `memberof-attr-name` setting. If you set it in the client configuration file, it is ignored.

To verify schema settings

1. From the Control Center, navigate to the **Preferences** view.
2. Validate the settings in the **Custom Unix Attributes** section.

Multi-schema handling

In VAS 3.5.*x*, you had to use the same schema for all forests in your domain. Authentication Services 4.*x* allows you to use different schemas for each forest in your domain.

Default user login name change

In VAS 3.5.*x*, the default user login name was the User Principal Name. However, Authentication Services 4.*x* uses the sAMAccountName as the default user login name.

To change the default user login name to the User Principal Name

1. From the Control Center, navigate to **Preferences | Custom Unix Attributes** and click **Customize**

2. Change the value in the **User Login Name** box to **userPrincipalName** and click **OK**.
3. In the **Confirm Schema Configuration Change** dialog, click **Yes**.

NOTE: See the *Authentication Services Installation Guide* for more information about how to use the Control Center.

Functionality changes

Functionality that you may be familiar with in VAS 3.5 has been changed.

Changes in VASTOOL output

Some `vastool` command output has changed in Authentication Services 4.*x*. Many error messages have been changed to be clearer and more informative. If you have scripts written to `vastool` you should test these scripts before rolling out an upgrade particularly if you parse `vastool` text output. Take special note of the following changes:

- `vastool checkaccess <user>` output was formatted as follows in 3.*x*:

  ```
  Access for service <service> by <user> is allowed.
  Access for service <service> by <user> is not allowed, <reason>.
  ```

- `vastool checkaccess <user>` output has been changed as follows in 4.*x*:

  ```
  ALLOWED [user=<user>] [service=<service>]
  DENIED (<reason>) [user=<user>] [service=<service>]
  ```

  This makes the result of the access check more obvious.

Internal database changes

Authentication Services 4.0 changed the format of the internal database. Thus, when upgrading from VAS 3.*x* to 4.1, all stored disconnected credentials become unusable and will be flushed. You will not have disconnected credentials until you have successfully logged in during a connected state.

vasfilter adm was removed

VAS 3.5 provided `vasfilter.adm` which allowed you to create limits on Unix values in the ADUC snap-in module. In Authentication Services 4.*x* you set the **Global Unix Options** in

the Control Center under **Preferences**.

PAM module changes

In VAS 3.5 the `pam` module was placed at the top of the PAM stack. In Authentication Services 4.*x* it is placed just before the local password validation module, usually `pam_unix`. When Authentication Services configures the PAM stack, it converts multi-line entries to one-line entries.

5

Upgrade the web console

In preparing for your Authentication Services upgrade, One Identity recommends that you install or upgrade Management Console for Unix first. This provides a mangement console that is a powerful and easy-to-use tool that dramatically simplifies deployment, enables management of local Unix users and groups, provides granular reports on key data and attributes, and streamlines the overall management of your Unix, Linux, and Mac OS X hosts.

- If you are upgrading from VAS 3.5, you must install Management Console for Unix for the first time. For more information, see Installing and configuring the management console on page 36.
- If you are upgrading from any Authentication Services 4.0 or above, you will be upgrading the console. For more information, see Upgrade Management Console for Unix 2.0 on page 40.

NOTE: Of course, you can install Authentication Services without using Management Console for Unix. For more information, see Upgrade Authentication Services client components manually on page 63. However, for the purposes of the examples in this guide, it is assumed that you will install and configure the Authentication Services Unix agent components by means of Management Console for Unix.

Installing and configuring the management console

The easiest way to install and configure Authentication Services Unix agent components is by means of Management Console for Unix.

NOTE: The procedures in this topic assume you do not have Management Console for Unix already installed.

To install the mangement console on a supported Windows platform

1. Mount the distribution media.

   Autorun starts automatically.

   NOTE: To start the Autorun installation wizard, you can also navigate to the root of the distribution media and double-click **autorun** Application file.

2. From the Authentication Services Autorun **Home** page, click the **Setup** tab.
3. From the **Setup** tab, click **One Identity Management Console for Unix**.

   The install wizard guides you through these setup dialogs:

   - **Management Console for Unix License Agreement** dialog
   - **Configure TCP/IP Port** dialog
   - **Installing** dialog

     Wait until it:

     - Extracts and installs Management Console for Unix on your computer.
     - Configures the database and service on the server.
     - Copies the Authentication Services client software packages for each platform.
     - Copies the Sudo Plugin software packages for each platform.
     - Copies the Privilege Manager for Unix Agent software packages for each platform.
     - Copies the Privilege Manager Policy Server packages for each platform.
   - **Completing the Management Console for Unix installation** dialog
4. In the **Complete** dialog, clear the **Launch the Management Console** option and click **Finish** to exit the install wizard and return to the Authentication Services Autorun **Setup** tab.

   Once you have installed Management Console for Unix, you are ready to install or upgrade the Authentication Services Windows components.

Upgrade Identity Manager for Unix 1.*x* web console

The process for upgrading the Web console from an older version is similar to installing it for the first time. The installer detects an older version of the console and automatically upgrades the components.

NOTE: The procedures in this topic assume you have Identity Manager for Unix 1.*x* installed.

Before you begin the upgrade procedure, close the Web console and make a backup of your database.

To upgrade the Web console

1. Backup the 1.0 database files:
   a. Shutdown the HSQLDB server.

      Management Console for Unix uses a HSQLDB (Hyper Structured Query Language Database) to store its data such as information about the hosts, settings, users, groups, encrypted passwords, and so forth.

   b. Copy the `/var/opt/quest/imu` data directory to a backup location.

      NOTE: Refer to *Appendix D: Database Maintenance* in the *One Identity Management Console for Unix Administration Guide* for more information about the database locations and filenames.

      Once you backup the database file, you are ready to start the upgrade.
2. Mount the Authentication Services 4.1 distribution media.

   Autorun starts automatically.

   NOTE: To start the Autorun installation wizard, you can also navigate to the root of the distribution media and double-click **autorun** Application file.
3. From the Authentication Services Autorun Home page, click the **Setup** tab.
4. From the *Setup* tab, click **Management Console for Unix**.
5. Click **Yes** when the installer detects an older version of the mangement console and asks if you want to continue.

   The install wizard guides you through the rest of the setup dialogs:

   - Management Console for Unix License Agreement
   - Configure TCP/IP Port
6. When the installer asks if you want to uninstall the previous version of the console, you can opt to leave the older version installed and continue the 2.*x* installation.

   NOTE:

   Once you are satisfied with the upgrade, you can uninstall 1.*x* at a later time.

   - On Windows, the Identity Manager for Unix Uninstaller is available from the *Start* menu at **Quest Software | Identity Manager for Unix**
   - On Unix, run the following command as root:

     `/opt/quest/imu/uninstall`
   - On Mac OS X, with root privileges, navigate to `/opt/quest/imu` and double-click **Identity Manager for Unix Uninstaller**.

   While you can have both the older and the newer versions of the mangement console installed, you can not run both at the same time.

7. In the **Complete** dialog, leave the **Launch the Management Console** option deselected and click **Finish** to exit the install wizard and return to the Authentication Services Autorun Setup tab.

   Once you have installed Management Console for Unix, you are ready to install or upgrade the Authentication Services Windows components.

8. After the upgrade, reassign Active Directory users to specific roles.

   The upgrade from 1.*x* to 2.*x* assigns any previously existing Active Directory to the **Manage Host** role. To assign Active Directory users to additional roles, navigate to **Preferences | System Settings | Roles and Permissions**. See *Add Role Members* in the mangement console online help for details.

NOTE:

After an upgrade from version 1.*x* to 2.*x*, please note the following:

- You must re-profile all managed hosts before you begin using the new features of Management Console for Unix.
- Because the encryption mechanism was changed, cached host credentials (that is, passwords cached by the ***supervisor*** account or Active Directory users with console access) are not migrated when you upgrade from 1.*x* to 2.*x*. Users will have to re-enter their passwords for hosts they manage the next time they perform tasks on the hosts and choose to cache them again on the server.
- The host address in the **Console host address** box on the *Console Information* settings may have been entered as a simple address in version 1.0. To perform some tasks in version 2.*x* without error, such as auto-profiling, the **Console host address** must be a Fully Qualified Domain Name.

Reset custom configuration settings

Reset custom configuration settings

When upgrading from version 1.0 to 2.*x* or higher, there are some steps you must take to reset any custom configuration settings you had in the previous version.

The upgrade procedure makes a `.bak` copy of your configuration file (`jvmargs.cfg.bak`) at the root of your installation directory. After you upgrade the mangement console from version 1.0 to 2.0, to reset any custom configuration settings you may have made in the previous version, compare the `jvmargs.cfg.bak` file with the new `jvmargs.cfg` file to see if you had any custom settings. For example, if you had increased the JVM Memory size in the previous version, then you will want to add the JVM Memory setting argument to the `custom.cfg` file. See *Overwriting Default Configuration Settings* in the mangement console online help for more information about customizing configuration settings for the mangement console.

NOTE: Do not change the `jvmargs.cfg` directly; the settings in the `custom.cfg` file overwrite the default settings in `jvmargs.cfg`.

By default, the installation directory is located at:

- On Windows 64-bit platforms:

  `%SystemDrive%:\Program Files\Quest Software\Management Console for Unix`

- On Windows 32-bit platforms:

  `%SystemDrive%:\Program Files (x86)\Quest Software\Management Console for Unix`

- On Unix/Mac OS X platforms:

  `/opt/quest/mcu`

Upgrade Management Console for Unix 2.0

The process for upgrading Management Console for Unix from an older version is similar to installing it for the first time. The installer detects an older version of the console and automatically upgrades the components.

NOTE: The procedures in this topic assume you have Management Console for Unix 2.0.x or greater installed.

Before you begin the upgrade procedure, close the console and make a backup of your database, as explained in step 1.

To upgrade Management Console for Unix

1. Backup the database files:
   a. Shutdown the service. See *Start/Stop/Restart Management Console for Unix Service* in the console online help for details.

      Management Console for Unix uses a HSQLDB (Hyper Structured Query Language Database) to store its data such as information about the hosts, settings, users, groups, and so forth.

   b. Copy the `/var/opt/quest/mcu` data directory to a backup location.

      Refer to *Database Maintenance* in the online help for more information about the database locations and filenames.

   c. After backup is complete restart the service. See *Start/Stop/Restart Management Console for Unix Service* in the console online help for details.

      Once you backup the database files, you are ready to start the upgrade.
2. To start the upgrade, follow the instructions for a first-time installation. See *Installing the Management Console* in the console online help for details.

   When the installer detects a previous version of the mangement console is already installed, it asks if you want to continue.
3. Click **Yes** in the **Install Management Console for Unix** dialog.
4. Accept the terms of the license agreement and click **Next**.

5. Modify the default SSL (https) and Non-SSL (http) port numbers, if necessary, and click **Install**.

   The installation wizard uninstalls the old version and configures the server database and service.

6. In the **Complete** dialog, select the **Launch the Management Console** option and click **Finish**.

NOTE: After an upgrade from any version of Management Console for Unix, it is important to re-profile all managed hosts.

6

Upgrade Authentication Services Windows components

One Identity recommends that you upgrade your Windows components before you upgrade the Unix components.

The process for upgrading the Authentication Services Windows components from older versions to version 4.1 is similar to the initial installation process. The Authentication Services Windows installer detects older versions and automatically upgrades them. The next time you launch Active Directory Users and Computers, you will see the updated Authentication Services property tabs.

NOTE: Have your license available for the Setup wizard.

Upgrading VAS 3.5 Windows components

You must install Authentication Services on all Windows Workstations you will use to administer Unix data in Active Directory.

To upgrade the Windows components

1. From the Authentication Services Autorun **Setup** tab, click **Authentication Services** to launch the Setup wizard.
2. Click **Yes** in the **Upgrade** dialog to indicate you want the wizard to uninstall the previous version of Vintela Authentication Services.

   The Authentication Services Setup Wizard starts automatically.
3. Click **Next** in the **Welcome** dialog and follow the wizard prompts.

   The wizard leads you through the following dialogs:

   - License Agreement
   - Choose Destination Location

- Ready to Install the Program
- InstallShield Wizard Complete

If you leave the **Launch Authentication Services** option selected in the **InstallShield Wizard Complete** dialog, when you click **Finish**, it detects if you have not configured Authentication Services for Active Directory and starts the Authentication Services Active Directory Configuration Wizard automatically. (Proceed to Configure Active Directory for Authentication Services on page 44 .)

Upgrading Authentication Services 4.*x* Windows components

If you had a previous version of the One Identity Identity Manager for Unix web console, upgrade to the Management Console for Unix mangement console to take advantage of the new features.

To upgrade the Authentication Services 4.x Windows components

1. From the Authentication Services Autorun *Setup* tab, click **Authentication Services** to launch the Setup wizard.

   The **InstallShield Wizard Welcome** dialog indicates that a previous installation was found.
2. Click **Next** in the **Welcome** dialog and follow the wizard prompts.

   The **Setup Status** dialog shows the progress of the upgrade:
   - Removing component registrations
   - Installing
   - Updating shortcuts
   - Registering components
3. In the **Update Complete** dialog, indicate whether you want to restart your computer now or later.

If you choose **No, I will restart my computer later**, the old version of the Control Center opens; you must restart your computer to complete the upgrade process.

7

Configure Active Directory for Authentication Services

To utilize full Active Directory functionality, when you install Authentication Services in your environment, One Identity recommends that you prepare Active Directory to store the configuration settings that it uses. Authentication Services adds the Unix properties of Active Directory users and groups to Active Directory and allows you to map a Unix user to an Active Directory user. This is a one-time process that creates the Authentication Services application configuration in your forest.

NOTE: To use the Authentication Services Active Directory Configuration Wizard, you must have rights to create and delete all child objects in the Active Directory container.

If you do not configure Active Directory for Authentication Services, you can run your Authentication Services client agent in Version 3 Compatibility Mode, which allows you to join a host to an Active Directory domain.

For more information, see Version 3 Compatibility Mode on page 48.

When running Authentication Services client agent in Version 3 Compatibility Mode, you have the option in One Identity Management Console for Unix to set the schema configuration to use Windows 2003 R2. See *Configure Windows 2003 R2 Schema* in the mangement console online help for details. The Windows 2003 R2 schema option extends the schema to support the direct look up of Unix identities in Active Directory domain servers.

You can also create the Authentication Services application configuration from the Unix command line, if you prefer. For more information, see *Creating the Application Configuration from the Unix Command Line* in the *Authentication Services Installation Guide*.

Configuring Active Directory for Authentication Services

The first time you install Authentication Services in your environment, One Identity recommends that you perform this one-time Active Directory configuration step to utilize full Authentication Services 4.1 functionality.

NOTE: If you do not configure Active Directory for Authentication Services, you can run your Authentication Services client agent in Version 3 Compatibility Mode, which allows you to join a host to an Active Directory domain.

For more information, see Version 3 Compatibility Mode on page 48.

To configure Active Directory for Authentication Services

1. In the **Authentication Services Active Directory Configuration Wizard Welcome** dialog, click **Next**.
2. In the **Connect to Active Directory** dialog:
   a. Provide Active Directory login credentials for the wizard to use for this task:
      - Select **Use my current AD logon credentials** if you are a user with permission to create a container in Active Directory.
      - Select **Use different AD logon credentials** to specify the Active Directory credentials of another user, enter the User name and Password.

      NOTE: The wizard does not save these credentials; it only uses them for this setup task.

   b. Indicate how you want to connect to Active Directory:

      Select whether to connect to an Active Directory Domain Controller or One Identity Active Roles Server.

      NOTE: If you have not installed the One Identity Active Roles Server MMC Console on your computer, the **ActiveRoles Server** option is not available.

   c. Optionally enter the domain or domain controller and click **Next**.
3. In the **License Authentication Services** dialog, browse to select your license file and click **Next**.

   Refer to Licensing Authentication Services on page 10 for more information about licensing requirements.

   NOTE: You can add additional licenses later in the **Authentication Services Control Center Preferences Licensing** dialog.
4. In the **Configure Settings in Active Directory** dialog, accept the default location in which to store the configuration or browse to select the Active Directory location

where you want to create the container and click **Setup**.

NOTE: You must have rights to create and delete all child objects in the selected location. For more information on the structure and rights required see Windows permissions on page 13.

5. Once you have configured Active Directory for Authentication Services, click **Close**.

   The Control Center opens. You are now ready to configure your Unix Agent Components.

   Proceed to Configure Unix agent components on page 50

About Active Directory configuration

The first time you install or upgrade the Authentication Services 4.1 Windows components in your environment, One Identity recommends that you configure Active Directory for Authentication Services to utilize full functionality. This is a one-time Active Directory configuration step that creates the application configuration in your forest. Authentication Services uses the information found in the application configuration to maintain consistency across the enterprise. Without the application configuration, store UNIX attributes in the RFC2307 standard attributes to achieve the most functionality.

NOTE: If you do not configure Active Directory for Authentication Services, you can run your client agent in Version 3 Compatibility Mode, which allows you to join a host to an Active Directory domain.

For more information, see Version 3 Compatibility Mode on page 48.

The Authentication Services application configuration stores the following information in Active Directory:

- Application Licenses
- Settings controlling default values and behavior for Unix-enabled users and groups
- Schema configuration

The Unix agents use the Active Directory configuration to validate license information and determine schema mappings. Windows management tools read this information to determine the schema mappings and the default values it uses when Unix-enabling new users and groups.

The Authentication Services application configuration information is stored inside a container object with the specific naming of: `cn={786E0064-A470-46B9-83FB-C7539C9FA27C}`. The default location for this container is `cn=Program Data,cn=Quest Software,cn=Authentication Services,dc=<your domain>`. This location is configurable.

There can only be one Active Directory configuration per forest. If Authentication Services finds multiple configurations, it uses the one created first as determined by reading the `whenCreated` attribute. The only time this would be a problem is if different groups were using different schema mappings for Unix attributes in Active Directory. In that case, standardize on one schema and use local override files to resolve conflicts. You can use the `Set-QasUnixUser` and `Set-QasUnixGroup` PowerShell commands to migrate Unix attributes

from one schema configuration to another. Refer to the PowerShell help for more information.

The first time you run the Control Center, the Authentication Services Active Directory Configuration Wizard walks you through the setup.

NOTE: You can also create the Authentication Services application configuration from the Unix command line, if you prefer.

For more information, see *Creating the Application Configuration from the Unix Command Line* in the *Authentication Services Installation Guide*.

You can modify the settings using the Authentication Services Control Center **Preferences**. To change Active Directory configuration settings, you must have rights to Create Child Object (container) and Write Attribute for `cn`, `displayName`, `description`, `showInAdvancedViewOnly` for the Active Directory configuration root container and all child objects.

In order for Unix clients to read the configuration, authenticated users must have rights to read `cn`, `displayName`, `description`, and `whenCreated` attributes for container objects in the application configuration. For most Active Directory configurations, this does not require any change.

The following table summarizes the required rights.

Table 14: Authentication Services: Required rights

Rights required	For user	Object class	Attributes
Create Child Object	Authentication Services Administrators Only	Container	`cn, displayName, description, showInAdvancedViewOnly`
Write Attribute	Authentication Services Administrators Only	Container	
Read Attribute	Authenticated Users	Container	`cn, displayName, description, whenCreated`

At any time you can completely remove the Authentication Services application configuration using the `Remove-QasConfiguration` cmdlet. However, without the application configuration, Authentication Services Active Directory-based management tools do not function.

Join the host to AD without the Authentication Services application configuration

You can install the Authentication Services Agent on a Unix system and join it to Active Directory without installing Authentication Services on Windows and setting up the Authentication Services Application Configuration.

The Authentication Services 4.x client-side agent required detection of a directory-based Application Configuration data object within the Active Directory forest in order to join the host computer to the Active Directory Domain. Authentication Services 4.0.2 removed this requirement for environments where directory-based User and/or Group identity information is not needed on the host Unix computer. These environments include full Mapped-User environments, GSS-API based authentication-only environments, or configurations where the Authentication Services agent will auto-generate posix attributes for Active Directory Users and Groups objects.

Version 3 Compatibility Mode

When upgrading to or installing Authentication Services 4.1, you can choose not to configure Active Directory for Authentication Services and run your Authentication Services client agent in Version 3 Compatibility Mode. While this prevents you from running the Control Center and accessing its many features and tools, you can join a host to an Active Directory domain when operating in Version 3 Compatibility Mode.

NOTE: When you run the `join` command without first creating a One Identity Application Configuration, Authentication Services displays a warning.

Without the Authentication Services application configuration the following information is stored locally:

- Application Licenses
- Settings controlling default values and behavior for Unix-enabled users and groups
- Schema configuration

Default user login name changes

In VAS 3.5.x, the default user login name was the User Principal Name; Authentication Services 4.1 uses the `sAMAccountName` as the default user login name. After upgrading to 4.1, if you want to continue to log in with the User Principal Name, then you must ensure that the `username-attr-name` in the `vas.conf` file is set to the User Principal Name before you begin the client agent upgrade.

NOTE: This is not necessary if the value of the User Principal Name prefix and the sAMAccountName are the same across your enterprise, which is the Active Directory default.

There are two ways to change the `username-attr-name` in the `vas.conf` file:

- Manually configure each client agent to use the User Principal Name
- Use Group Policy to automatically configure all the clients in your environment

To manually configure each client agent to use the User Principal Name

1. Before you upgrade each client agent, open the `/etc/opt/quest/vas/vas.conf` file and find the `username-attr-name` attribute in the `[vasd]` section.
2. If there is no value set for this attribute, then set it to:

   ```
   username-attr-name = userPrincipalName
   ```

   NOTE: If the attribute is already explicitly set to another value (such as: `username-attr-name = uid`), do not change it.

   Alternatively, you can run the following command on each client to change the setting in `vas.conf`:

   ```
   vastool configure vas vasd username-attr-name userprincipalname
   ```

To automatically configure all the clients in your environment

1. Open the **Group Policy Management Editor** and navigate to **Computer Configuration | Policies | Unix Settings | Authentication Services | Client Configuration**.

   NOTE: Your version of Group Policy Management Editor may not have the **Policies** directory layer.
2. Double-click **Authentication Services Configuration** to open **Properties**.
3. Enter **username-attr-name** in the **vas.conf Settings** box and click **Search**.
4. Enter **userPrincipalName** and click **OK**.

Best practice

Because Version 3 Compatibility Mode does not allow you run the Control Center and access its many features and tools, One Identity recommends that you create the application configuration so you can utilize full Authentication Services 4.1 functionality.

There are two ways to create the application configuration:

1. When you start the Control Center from a Windows workstation, the **Set up Authentication Services Active Directory Configuration Wizard** starts automatically to lead you through the process of configuring Active Directory for Authentication Services.
2. Alternatively, you can run `vastool configure ad` from the Unix command line to create the One Identity Application Configuration in Active Directory.

8

Configure Unix agent components

The Control Center gives you access to the tools you need to perform Unix identity management tasks.

NOTE: If the Control Center is not currently open, you can either double-click the desktop icon or access it by means of the **Start** menu.

Follow the steps outlined on the Control Center **Home** page to get your Unix agents ready.

NOTE: Of course, you can install Authentication Services without using Management Console for Unix. You can find those instructions in the *Installing and Joining from the Unix Command Line* section of the *Authentication Services Installation Guide*, located in Control Center **Tools** view or in the docs directory of the installation media. However, for the purposes of the examples in this guide, it is assumed that you will install and configure the Authentication Services Unix agent components by means of Management Console for Unix.

To start the mangement console

1. From the Control Center, click the **Management Console** link in the left navigation pane.

Set up Management Console for Unix wizard

The first time you launch the mangement console, the **Setup One Identity Management Console for Unix** wizard leads you through some post-installation configuration steps.

Choose one of these options:

- **Skip the Active Directory configuration, I'll do that later from the console**

  This option allows you to use the core features of the console and limits access to the console to the default **supervisor** account only.
- **Walk me through the configuration steps for using AD user accounts for logon to the console**

When you configure the console for Active Directory, you unlock additional Active Directory features.

NOTE: To use the mangement console with Authentication Services, or to use roles to allow access to the console using Active Directory, you must configure the console for Active Directory log on.

Choose an option and click **Next**.

NOTE: If you choose the **Skip** option, the **Identify Console** dialog displays. For more information, see Identify Console dialog on page 52.

If you choose the **Walk me through** option, it allows you to configure the console for Active Directory log on. See *Configure the Console for Active Directory* in the *One Identity Management Console for Unix Administration Guide* for details.

NOTE: If you can not configure the console for Active Directory during your initial installation of Management Console for Unix, choose the **Skip** option. After the installation, log into the console as **supervisor** and configure the console for Active Directory from System Settings. See *Active Directory Configuration* in the *One Identity Management Console for Unix Administration Guide* for more information.

Configure Console for Active Directory Logon dialog

The **Setup Management Console for Unix** wizard opens the **Configure Console for Active Directory Logon** dialog when you choose the **Walk me through the configuration steps for using AD user accounts for logon to the console** option.

To configure the mangement console for Active Directory logon

1. In the **Configure Console for Active Directory Logon** dialog, enter a valid Active Directory domain in the forest, in the form **example.com**.
2. Enter the credentials for an Active Directory account that has logon rights.

   Enter a sAMAccountName, which uses the default domain or a User Principal Name, as in **username@domain**. The wizard uses these credentials to configure the mangement console for use with Active Directory.

   NOTE: This is a read-only operation; no changes are made to Active Directory.
3. Click **Connect to Active Directory**.
4. When you see the message that indicates the console connected to Active Directory successfully, click **Next**.

   The **Set up console access by role** dialog opens.

Set up console access by role dialog

After you configure the console for Active Directory logon, the setup wizard displays the **Set up console access by role** dialog.

To add Active Directory users or groups to the console access list

1. In the **Set up console access by role** dialog, click **Add** to specify the Active Directory users and groups that you want to have access to the features available in Management Console for Unix.
2. In the **Select Users and Groups** dialog, use the search controls to find and select Active Directory users or groups. Select one or more objects from the list and click **OK**.

   The mangement console adds the selected objects to the list in the **Set up console access by role** dialog.

   By default the mangement console assigns users to **All Roles**, which gives those accounts permissions to access and perform all tasks within the console. See *Console Roles and Permissions System Settings* in the *One Identity Management Console for Unix Administration Guide* for details.
3. Click in the **Roles** cell to activate a drop-down menu from which you can choose a role for the user account.

   NOTE: During the initial setup, you can only assign one role per user. Add additional roles to a user in **System Settings**. See *Add (or Remove) Role Members* in the *One Identity Management Console for Unix Administration Guide* for details.
4. Click **Next** to save your selections.

   The **Identify Console** dialog opens.

Identify Console dialog

The setup wizard displays the **Identify Console** dialog during the post-installation configuration steps. The Authentication Services Control Center uses this information to identify this mangement console. Hosts configured for automatic profiling pr automatic QAS agent status also use this information to contact the mangement console server.

To identify the mangement console

1. In the **Identify Console** dialog, modify the information about this mangement console, if necessary, and click **Next** to open the **Set supervisor password** dialog.

   NOTE: You can modify these settings from **Settings | System settings | General | Console Information**. See *Console Information Settings* in the console's online help for details.

Set Supervisor Password dialog

The **supervisor** account is the default account for accessing all features of the mangement console. The **supervisor** is a member of all roles and no permissions can be removed from **supervisor**. However, the **supervisor** does not have Active Directory credentials and therefore is blocked from performing Active Directory tasks.

To set the supervisor password

1. In the **Set supervisor password** dialog, enter a password for the **supervisor** account and click **Next**.

   The **Summary** dialog displays.

2. To log on using the console supervisor account, use **supervisor** as the user name.

NOTE: The **supervisor** is the only account that has rights to change the **supervisor** account password in System Settings. See *Reset the Supervisor Password* in the mangement console online help for details.

Summary dialog

To complete the Management Console for Unix Setup wizard

1. In the **Summary** dialog, click **Finish**.

   The Management Console for Unix login screen opens.

Logging in to Management Console for Unix

Whenever you launch the mangement console, you must enter an authorized account to proceed. The Management Console for Unix features that are available depend on the account with which you log in.

To use the core version to manage local Unix users and groups and to access the mangement console system settings, you must use the **supervisor** account (that is, you must log on with the **supervisor** user name). However, to use the Active Directory features of Management Console for Unix, you must log on with an Active Directory account that has been granted access to the mangement console, that is, defined during the post-installation configuration. See *Setup Console Access by Role* in the online help for details. To add additional accounts to this access list, see *Add (or Remove) Role Members* in the online help for details.

To log on to the mangement console

1. Enter the user name and password and click **Sign In**.

   Enter:

   - The **supervisor** account name
   - A sAMAccountName, which uses the default domain
   - A User Principal Name in the form, username@domain

   The mangement console opens and displays the user name you specified in the upper right-hand corner of the screen.

2. To log on using a different account, click the authenticated user's login name and click **Sign Out**. Then sign back on using a different account.

   The **Log-on** page redisplays, allowing you to enter a different account.

Prepare Unix hosts

The mangement console provides a central management and reporting console for local Unix users and groups.

Using Management Console for Unix with Authentication Services not only allows you to centrally manage your hosts, but it allows you to do these additional features for managing Unix systems with Active Directory:

- Ability to remotely install Authentication Services agents, join systems to Active Directory, and implement AD-based authentication for Unix, Linux, and Mac OS X systems.
- Ability to manage access control on a single host system or across multiple hosts.
- Ability to create reports about Unix-enabled users and groups in Active Directory.
- Ability to create access control reports that show which user is permitted to log into which Unix host.

Whether you have the core version or are using the mangement console with Authentication Services, once you have successfully installed Management Console for Unix, you must first add your hosts to the console, and then profile them to gather system information. Once a host is added and profiled you can then manage users and groups on the hosts and run reports.

Adding hosts to the management console

In order to manage a Unix host from the mangement console, you must first add the host. Go to the **Hosts** tab of the mangement console to either manually enter hosts or import them from a file.

To add hosts to the mangement console

1. Click the **Add Hosts** tool bar button to display the **Add Hosts** dialog.
2. To manually add one or more hosts, enter the FQDN, IP address, or short name of a host you want to add to the mangement console and either click the **Add** button or press **Enter**.

   Once added, the **Host** column displays the value you enter. The mangement console uses that value to connect to the host. You can rename the host if it has not been profiled using the **Rename Host** command on the **Host** panel of the tool bar. After a host is profiled, the only way to change what is displayed in the **Host** column is to remove the host from the console and re-add it. For example, if you add a host by its IP address, the IP address displays in the **Host** column (as well as in the **IP Address** column); to change what is displayed in the **Host** column, you must use the **Remove from console** tool bar button to remove the host from the console; then use the **Add Hosts** button to re-add the client by its host name. If you had profiled the host before removing it, you will have to re-profile it after re-adding it.
3. To add hosts from a `known_hosts` file, click the **Import** button.
   a. In the **Import hosts from file** dialog, browse to select a `.txt` file containing a list of hosts to import.

      Once imported, the host addresses display in the **Add Host** dialog list.

      NOTE: The valid format for an import file is:

      - `.txt` file - contains the IP address or DNS name, one per line
      - `known_hosts` file - contains address algorithm hostKey (separated by a space), one entry per line

      See *Known_hosts File Format* in the online help for more information about the supported known_hosts file format.
4. Once you have a list of one or more hosts to add, if you do not wish to profile the hosts at this time, clear the **Profile hosts after adding** option.

   NOTE: If you add more hosts to the list than selected in the **Rows to show** drop-down menu in the **View** panel of the tool bar, this option is disabled.
5. If you do not clear the **Profile hosts after adding** option in the **Add Hosts** dialog, when you click **OK**, the **Profile Host** dialog prompts you to enter the user credentials to access the hosts. Refer to Profiling hosts on page 56, which walks you through the host profile steps.
6. If you clear the **Profile hosts after adding** option in the **Add Hosts** dialog, when you click **OK**, the **Add Hosts** dialog closes and control returns to the mangement console.

   The mangement console lists hosts that were successfully added on the **All Hosts** view by the FQDN, IP address, or short name of the hosts you entered in the **Add Hosts** dialog.

Profiling hosts

Profiling imports information about the host, including local users and groups, into the mangement console. It is a read-only operation and no changes are made to the host during the profiling operation. Profiling does not require elevated privileges.

To profile hosts

1. Select one or more hosts in the **All Hosts** view and click **Profile** from the **Prepare** panel of the tool bar, or open the **Profile** menu and choose **Profile**.
2. In the **Profile Host** dialog, enter user credentials to access the hosts.

   If you selected multiple hosts, you are asked if you want to use the same credentials for all the hosts (default) or enter different credentials for each host.
3. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter the following information:
   a. Enter the user name and password to log onto the selected hosts.
   b. Optionally, enter the SSH port to use. It uses port 22 by default.
   c. To save the credentials entered for the host, select the **Save my credentials on the server** option.

      Once saved, the mangement console uses these credentials to access the host during this and subsequent sessions.

   NOTE: If you do not save a password to the server, the user name and password fields will be blank the first time the mangement console needs credentials to complete a task on the host during a logon session. Once entered, the mangement console caches the user name and password and reuses these credentials during the current session, and pre-populates the user name and password fields in subsequent tasks during the current log on session.

   If you choose to save a host's credentials to the server, the mangement console encrypts the credentials and saves them in the Java keystore. Saved user names and passwords persist across logon sessions, and when needed, the mangement console pre-populates the user name and password fields each subsequent time it needs them to perform a task. For more information, see *Caching Unix Host Credentials* in the online help.
4. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays allowing you to enter different credentials and specify different settings for each host.
   a. To enter different credentials, place your cursor in the **Username** and **Password** columns to the right of the **Host** column and enter the credentials to use.
   b. To change the SSH port for a host, place your cursor in the **SSH Port** column and enter the new SSH port number.

c. To save the credentials entered for a host, select the check box in the **Save** column.

5. If you want the mangement console to prompt you to review and accept new SSH keys for the selected hosts (which do not have previously cached SSH keys), clear the **Automatically accept SSH keys** option before you click **OK**.

   NOTE: When profiling one or more hosts, you must accept at least one key before continuing. The mangement console only profiles hosts with accepted keys.

   By default, the **Automatically accept SSH keys** option is selected. This enables the mangement console to automatically accept the SSH key for all selected hosts that do not have a previously cached key. When it accepts the key, the console adds it to the accepted-keys cache on the Management Console for Unix server. If you clear the **Automatically accept SSH keys** option, when the mangement console encounters a modified key, it opens the **Validate Host SSH Keys** dialog, allowing you to manually accept keys that are encountered. Once you have manually verified the fingerprint, the console adds the SSH host keys to the accepted-keys cache.

   NOTE: Once you profile a host, all future tasks that involve an SSH connection will verify the SSH host key against the accepted-keys cache. When profiling, if the console encounters a modified key, the profile task prompts you to accept and new or changed keys. When performing any other SSH action, other than profile, if the console encounters a different SSH key, the task will fail. To update the accepted-keys cache for the host, you can either profile or reprofile the host, accept the new key, and try the task again. Or, you can import a new SSH host key from the host's properties or from the **All Hosts** view. See *Import SSH Host Key* or *Managing SSH Host Keys* in the online help for more information.

A progress bar displays in the **Task Progress** pane. The final status of the task displays, including any failures or advisories encountered.

Configuring automatic profiling

To keep the Management Console for Unix database up to date with accurate information about users, groups, and One Identity products, you can configure the mangement console to profile hosts automatically.

BEST PRACTICE: Configure newly added hosts for auto-profiling before you perform any other actions so that the mangement console dynamically updates user and group information. See *UID or GID Conflicts* in the online help.

Configuring a host for auto-profiling sets up a cron job on the client that runs every five minutes. If it detects changes on the host, it triggers a profile operation.

The cron job detects changes to the following:

- Local users, groups, or shells
- Installed Authentication Services or Privilege Manager software

- Authentication Services access control lists
- Authentication Services mapped user information
- Privilege Manager configuration
- Authentication Services configuration
- Privilege Manager licenses

The cron job also sends a heartbeat every day. This updates the **Last profiled** date displayed on the host properties. If the **Last profiled** date is more than 24 hours old, the host icon changes to  to indicate no heartbeat.

To configure automatic profiling

1. Select one or more hosts in the **All Hosts** view, open the **Profile** menu from the **Prepare** panel of the tool bar, and choose **Profile Automatically**.

   NOTE: The **Profile Automatically** option is only available for multiple hosts if all hosts are in the same "auto-profile" state; that is, they all have **Auto-profiling** turned on, or they all have **Auto-profiling** turned off.

2. In the **Profile Automatically** dialog, select the **Profile the host automatically** option.
3. Choose the user account you want to use for profiling:
   - **Create a user service account on the host**

     When you choose to create the user service account on the host, if it does not already exist, the mangement console, does the following:

     a. Creates "questusr," the user service account, and a corresponding "questgrp" group on the host that the mangement console uses for automatic profiling.

     b. Adds *questusr* as an implicit member of *questgrp*.

     -OR-

   - **Use an existing user account (user must exist on all selected hosts)**

     Click **Select** to browse for a user.
4. Click **OK** in the **Profile Automatically** dialog.

   Whether you choose to create the user service account or use an existing user account, the mangement console:

   - Adds the user account (the "questusr" or your existing user account) to the `cron.allow` file, if necessary. For example, the console takes no action if the `cron.allow` file does not already exist, but there is a `cron.deny` file:

cron.allow	cron.deny	Console's action	Resultant user access
NO	NO	Creates `cron.allow` and adds root and *questusr* to it	Both root and *questusr* have access.
NO	YES	No action	All users have access except those in `cron.deny`; *questusr* has access unless explicitly denied.
YES	NO	Adds *questusr* to `cron.allow`	Users in `cron.allow` have access.
YES	YES	Adds *questusr* to `cron.allow`	Users in `cron.allow` have access unless in `cron.deny`.

- Adds the auto-profile SSH key to *questusr*'s authorized_keys, `/var/opt/quest/home/questusr/.ssh/authorized_keys`.
- Verifies the service account user can log in to the host.

NOTE: If you receive an error message saying you could not log in with the user service account, please refer to *Service Account Login Fails* in the online help to troubleshooting this issue.

The *questusr* account is a non-privileged account that does not require root-level permissions. This account is used by the console to gather information about existing user and groups in a read-only fashion; however, the mangement console does not use *questusr* account to make changes to any configuration files.

If *questusr* is inadvertently deleted from the console, the console turns auto-profiling off.

To re-create the "questusr" account

a. Re-profile the host.

b. Reconfigure the host for automatic profiling.

5. In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

NOTE: This task requires elevated credentials.

If you select multiple hosts, you are asked if you want to use the same credentials for all the hosts (default) or enter different credentials for each host.

a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.

b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid is displayed that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

To disable automatic profiling

1. Select one or more hosts on the **All Hosts** view and choose **Profile Automatically**.
2. Clear the **Profile the host automatically** option and click **OK**.
3. In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

When you disable auto-profiling for a host, the mangement console:

1. Leaves the "questusr" and the corresponding "questgrp" accounts on the host, if they were previously created.
2. Leaves *questusr* as an implicit member of *questgrp*, if it exists.
3. Removes the user account (the "questusr" or your existing user account) from the `cron.allow` file.
4. Removes the auto-profile SSH key from that user's `authorized_keys` file.

Checking readiness

Once you install the software on your remote hosts, the mangement console allows you to perform a series of tests to verify that a host meets the minimum requirements to join an Active Directory domain. Running the readiness checks does NOT require elevated privileges.

NOTE: This task is only available when you are logged on as ***supervisor*** or an Active Directory account in the Manage Hosts role. See *Roles and Permissions System Settings* in the mangement console online help for more information.

To check hosts for Active Directory Readiness

1. Select one or more hosts on the **All Hosts** view of the **Hosts** tab, open the **Check** menu from the **Prepare** panel of the tool bar, and choose **Check for AD Readiness**.
2. In the **Check AD Readiness** view, enter the Active Directory domain to use for the readiness check.
3. Enter Active Directory user credentials, and click **OK**.
4. In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

   If you selected multiple hosts, it asks whether you want to use the same credentials for all the hosts (default) or enter different credentials for each host.

a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.

b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

A progress bar displays in the **Task Progress** pane on the **All Hosts** page. The final status of the task displays, including any failures or advisories encountered. To see the AD Readiness check results, open the host's property page and select the **Readiness Check Results** tab.

Installing software on hosts

Once you have successfully added and profiled one or more hosts, and checked them for AD Readiness, you can remotely deploy software products to them from the mangement console.

To install Authentication Services software on hosts

1. Select one or more profiled hosts on the **All Hosts** view and click the **Install Software** tool bar button.

   NOTE: The **Install Software** tool bar menu is enabled when you select hosts that are profiled.

   The tool bar button will not be active if:

   - You have not selected any hosts.
   - You have selected multiple hosts with different states (added, profiled, or joined).

2. In the **Install Software** dialog, select the Authentication Services software products you want to install and click **OK**.

   - **Authentication Services Agent (Required)**: Select to allow Active Directory users access to selected host. Authentication Services provides centralized user and authentication management. It uses Kerberos and LDAP to provide secure data transport and an authentication framework that works with Microsoft Active Directory. Components include `vasd`, `nss_vas`, `pam_vas`, and `vastool`.
   - **Authentication Services for Group Policy (Required)**: Select to install the Group Policy component that provides Active Directory Group Policy support for Unix, Linux, and Mac OS X platforms.
   - **Authentication Services for NIS**: Select to install the NIS Proxy component that provides the NIS compatibility features for Authentication Services. `vasyp` is a NIS daemon that acts as a `ypserv` replacement on each host.

- **Authentication Services for LDAP**: Select to install the LDAP Proxy component that provides a way for applications that use LDAP bind to authenticate users to Active Directory without using secure LDAP (LDAPS). Instead of sending LDAP traffic directly to Active Directory domain controllers, you can configure applications to send plain text LDAP traffic to `vasldapd` by means of the loopback interface. `vasldapd` proxies these requests to Active Directory using Kerberos as the security mechanism.
- **Dynamic DNS Updater**: Select to install the Dynamic DNS Updater component that provides a way to dynamically update host records in DNS and can be triggered by DHCP updates.
- **Defender PAM Module**: Select to install the Defender authentication components for PAM based Unix/Linux systems. Includes PAM module, documentation, and utilities to appropriately configure the PAM subsystem for Active Directory/Defender OTP authentication.

NOTE: You must install the Authentication Services Agent and the Group Policy packages.

NOTE: If you do not see all of these software packages, verify the path to the software packages is correctly set in **System Settings**. Refer to *Set the Authentication Services Client Software Location on the Server* in the mangement console online help for details.

3. In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

   NOTE: This task requires elevated credentials.

   If you selected multiple hosts, it asks whether you want to use the same credentials for all the hosts (default) or enter different credentials for each host.

   a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.

   b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

Upgrade Authentication Services client components manually

The easiest way to upgrade Authentication Services client components is from Management Console for Unix. Once a you have successfully added and profiled one or more hosts, you can remotely deploy software products to them from the mangement console. For more information, see Configure Unix agent components on page 50.

You can also upgrade your Authentication Services client components from the Unix command line, if you prefer.

Upgrading VAS 3.5 from the command line

To upgrade VAS 3.5 from the Unix command line

1. Install the upgrade package on that host by running:

   ```
   # ./install.sh upgrade
   ```

   NOTE: If you are running your client agent in *Version 3 Compatibility Mode*, Authentication Services displays a warning message. For more information, see Version 3 Compatibility Mode on page 48.

2. Install the Authentication Services license. See Licensing Authentication Services on page 10.
3. Create the Authentication Services application configuration. See *Creating the Application Configuration from the Unix Command Line* in the *Authentication Services Installation Guide* for more information.

   NOTE: This step is optional. If you do not configure Authentication Services for Active Directory, you can run your Authentication Services client agent in "Version 3 Compatibility Mode" which allows you to join a host to an Active Directory domain.

4. Upgrade the rest of your hosts to the Authentication Services 4.1 Agent.

About the Authentication Services Application Configuration

The first time you install or upgrade the Authentication Services 4.1 Windows components in your environment, One Identity recommends that you configure Active Directory for Authentication Services to utilize full Authentication Services 4.1 functionality. This is a one-time Active Directory configuration step that creates the Authentication Services application configuration in your forest. Authentication Services uses the information found in the application configuration to maintain consistency across the enterprise.

If you upgrade VAS 3.5 to Authentication Services 4.1 using Management Console for Unix as explained in the *Authentication Services Upgrade Guide*, the Authentication Services Active Directory Configuration Wizard starts automatically to assist you in setting up the application configuration; however, if you are upgrading from the Unix command line, you can create the Authentication Services application configuration using the `vastool` command.

NOTE: You need only one application configuration per forest. If you already have an Authentication Services application configuration in your forest, you do not need to create another one. For more information, see About Active Directory configuration on page 46.

Authentication Services agent upgrade commands

To upgrade the Authentication Services agent package

1. Log in and open a root shell.
2. Mount the installation DVD and run the appropriate command.

   See Additional configuration information that follows the table.

Table 15: Authentication Services: Agent upgrade commands

Platform	Command
Linux x86 - RPM	# rpm -Uhv /<*mount*>/client/linux-x86/vasclnt-<*version*>-<*build*>.i386.rpm
Linux x64 - RPM	# rpm -Uhv /<*mount*>/client/linux-x86_64/vasclnt-<*version*>-<*build*>.x86_64.rpm
Linux x86 - DEB	# dpkg -i /<*mount*>/client/linux-x86/vasclnt-<*version*>-<*build*>.i386.deb
Linux x64 - DEB	# dpkg -i /<*mount*>/client/linux-x86_64/vasclnt-<*version*>-<*build*>_amd64.deb
Linux s390	# rpm -Uhv /<*mount*>/client/linux-s390/vasclnt-<*version*>-<*build*>.s390.rpm

Platform	Command
Linux s390x	# rpm -Uhv /<*mount*>/client/linux-s390x/vasclnt-<*version*>-<*build*>.s390x.rpm
VMware ESX 3.*x*	# rpm -Uhv /<*mount*>/client/linux-x86/vasclnt-<*version*>-<*build*>.i386.rpm
VMware ESX 4.1	# rpm -Uhv /<*mount*>/client/linux-x86_64/vasclnt-<*version*>-<*build*>.x86_64.rpm
SLES 8 PPC	# rpm -Uhv /<*mount*>/client/linux-glibc22-ppc64/vasclnt-glibc22-<*version*>-<*build*>.ppc64.rpm
SLES 9 PPC	# rpm -Uhv /<*mount*>/client/linux-glibc23-ppc64/vasclnt-glibc23-<*version*>-<*build*>.ppc64.rpm
Solaris 8-10 x86	# pkgadd -d /<*mount*>/client/solaris8-x86/vasclnt_SunOS_5.8_i386-<*version*>-<*build*>.pkg vasclnt
Solaris 10 x64	# pkgadd -d /<*mount*>/client/solaris10-x64/vasclnt_SunOS_5.10_i386-<*version*>-<*build*>.pkg vasclnt
Solaris 8-10 SPARC	# pkgadd -d /<*mount*>/client/solaris8-sparc/vasclnt_SunOS_5.8_sparc-<*version*>-<*build*>.pkg vasclnt
HP-UX PA-RISC 11i v1 (B.11.11)	# swinstall -s /<*mount*>/client/hpux-pa/vasclnt_9000-<*version*>-<*build*>.depot vaslcnt
HP-UX PA-RISC 11i v2 (B.11.23), 11i v3 (B.11.31)	# swinstall -s /<*mount*>/client/hpux-pa-11v1/vasclnt_hpux-11.11-<*version*>-<*build*>.depot vasclnt
HP-UX IA64 11i v1.6 (B.11.22), 11i v2 (B.11.23), 11i v3 (B.11.31)	# swinstall -s /<*mount*>/client/hpux-ia64/vasclnt_ia64-<*version*>-<*build*>.depot vasclnt
AIX 4.3.3	# installp -acXd /<*mount*>/client/aix-43/vasclnt.AIX_4.3.<*version*>-<*build*>.bff all
AIX 5.1 – 5.2	# installp -acXd /<*mount*>/client/aix-51/vasclnt.AIX_5.1.<*version*>-<*build*>.bff all
AIX 5.3 – 6.1	# installp -acXd /<*mount*>/client/aix-53/vasclnt.AIX_5.3.<*version*>-<*build*>.bff all
Mac OS X	/usr/sbin/installer -pkg '/<mount>/VAS.mpkg/Contents/Packages/vasclnt.pkg' -target /

Additional configuration information

NOTE: During the upgrade, `vasd` reloads and updates its user and group cache. To restart the Authentication Services caching service, see Restarting Authentication Services services on page 67.

NOTE: **VMware**: VMware provides a Host Update Utility to upgrade an ESX 3.5 agent to 4.0, but if Authentication Services is left installed and configured during the procedure, the machine will be inaccessible after the upgrade. This is because the previous 3.5 installation is pushed aside and mounted under the `/esx3-installation` directory, but all the key configuration files, like `/etc/nsswitch.conf` and the `pam.d` directory, are preserved.

If Authentication Services is still configured in those files, it leaves the machine in a bad state. Because of this, One Identity recommends that you uninstall Authentication Services before attempting to upgrade to ESX 4.0. In the *vSphere Upgrade Guide*, VMware warns that "no third-party management agents or third-party software applications are migrated," but it does not explicitly say they should be uninstalled prior to upgrade.

Should you accidentally leave Authentication Services installed or configured during the upgrade, use the following steps to fix the machine:

1. Boot into single user mode.
2. Copy `/etc/pam.d/vmware-authd.esx4` over `/etc/pam.d/vmware-authd` (backup `vmware-authd` first if desired).
3. Copy `/etc/pam.d/system-auth-generic.esx4` over `/etc/pam.d/system-auth-generic`.
4. Remove "vas4" from the passwd, group, and any other configured lines in `nsswitch.conf`
5. Reboot the machine--the machine should now be accessible.
6. Install the `linux-x86_64`Authentication Services packages.

NOTE: **Solaris**: The `-a vasclient-defaults` option specifies an alternative default file for `pkgadd` administrative options that allows `pkgadd` to overwrite an existing package with a new package.

`pkgadd` does not support the concept of upgrading a package, so this allows you to upgrade without having to rejoin your machine to the Active Directory domain, or uninstalling the old version first.

NOTE: **HP-UX**: Reboot the HP-UX machine to ensure that all of the new files are installed. HP-UX does not allow you to overwrite files that are in use—this is done as part of the boot sequence.

Restarting Authentication Services services

1. The method for restarting services varies by platform:
    a. To restart Authentication Services on Linux or Solaris, enter:

    ```
    /etc/init.d/vasd restart
    ```

    b. To restart Authentication Services on HP-UX, enter:

    ```
    /sbin/init.d/vasd restart
    ```

    c. To restart Authentication Services on AIX, enter:

    ```
    stopsrc -s vasd
    startsrc -s vasd
    ```

NOTE: Due to library changes between the Authentication Services 3.*x* and 4.1, One Identity recommends that you restart all long-lived processes that use Authentication Services data to force a reload of the newer libraries. For example, you must restart `cron`.

Getting started with Authentication Services

Once you have successfully installed Authentication Services, you will want to learn how to do some basic system administration tasks using the Control Center and Management Console for Unix.

Getting acquainted with the Control Center

Authentication Services consists of plugins, extensions, security modules, and utilities spread across nearly every operating system imaginable. The Control Center pulls those parts together and provides a single place for you to find the information and resources you need.

Control Center installs on Windows and is a great starting place for new users to get comfortable with some of Authentication Services' capabilities.

You can launch the Control Center from the *Start* menu or by double-clicking the desktop icon, or by double-clicking the Control Center application file from `%SystemDrive% :\Program Files (x86)\Quest Software\Authentication Services`.

Table 16: Control Center: Navigation links

Control Center Section	Description
Home	The **Welcome** page provides information about how to use the Control Center tools and features.
Management Console	You can run the One Identity Management Console for Unix mangement console within the Control Center or you can run it separately in a supported web browser. The mangement console is a separate install on Windows, Unix, Linux, or Mac OS X that you can launch from the ISO.

Control Center Section	Description
	Typically, you install one mangement console per environment to avoid redundancy. One Identity does not advise managing a Unix host by more than one mangement console in order to avoid redundancy and inconsistencies in stored information. If you manage the same Unix host by more than one mangement console, you should always re-profile that host to minimize inconsistencies that may occur between instances of the mangement consoles.
Group Policy	The Control Center provides the ability to search on Active Directory Group Policy Objects that have Unix and Mac OS X settings defined. Also provides links to edit these GPOs and run reports that show the detailed settings of the Group Policy Objects.
Tools	The Control Center provides links to additional tools and resources available with Authentication Services. A great starting place for anyone new to the product.
Preferences	The Control Center allows you to centrally manage the default values generated by the various Authentication Services management tools, including the ADUC snap-in, the PowerShell cmdlets, and the Unix command-line tools.
Log into remote host	The Control Center provides a simple SSH client (built on PuTTY) for remote access to Unix systems; simplifies new installs from having to find and install a separate PuTTY client.

To run the Control Center, you must be logged in as a domain user. To make changes to global settings, you must have rights in Active Directory to create, delete, and modify objects in the Authentication Services configuration area of Active Directory.

Management console

Management Console for Unix allows you to centrally manage Authentication Services agents running on Unix, Linux, and Mac OS X systems.

With the mangement console you can:

- Remotely deploy the Authentication Services agent software.
- Manage local user and group accounts.
- Configure account mappings from local users to Active Directory accounts.
- Report on a variety of security and host access related information.

You can install the mangement console on supported Unix, Linux, and Mac OS X platforms. Once installed, you can access it from a browser using default port of 9443 or from the Control Center.

Group Policy

Microsoft Group Policy provides excellent policy-based configuration management tools for Windows. Group Policy allows you to manage Unix resources in much the same way. Group Policy allows you to consolidate configuration management tasks by using the Group Policy functionality of Microsoft Windows Server to manage Unix operating systems and Unix application settings.

To open Group Policy, click **Group Policy** on the left navigation panel of the Authentication Services Control Center.

Filtering the list of GPOs

To filter the list of GPOs

1. Expand the **Filter Options** section.
2. Enter all or part of a name to filter the list of GPOs.
3. Open the **Domain** drop-down menu to choose a domain.
4. Select the **Unix Settings** or **Mac Settings List Only** options to further filter the GPO list.

   If you select both options, only the GPOs configured for both Unix and Mac OS X display.

Editing a GPO

To edit a group policy object

1. From the **Group Policy** window, select a GPO in the list and click **Edit GPO** from the **Actions** menu.

   The **Group Policy Object Editor** opens for the selected GPO.

   NOTE: For more information about the group policies, refer to the *Authentication Services Administration Guide*, located in Control Center **Tools** view in the **Documentation** section, or in the `docs` directory of the installation media.

Generating a settings report

A settings report displays all of the Authentication Services Group Policy object settings that apply to Unix or Mac OS X systems.

To generate a settings report

1. From the window, select a GPO Name and click **Settings Report** from the **Actions** menu.

   An HTML report of the currently configured Unix and Mac OS X settings displays.

   NOTE: You can select multiple GPOs to run several reports simultaneously.

Showing files

To open the Windows Explorer

1. From the **Group Policy** window, select a GPO in the list and click **Show Files** from the **Actions** menu.

   The Windows Explorer opens and displays the Group Policy Templates for the selected GPO.

Launching GPMC

NOTE: Microsoft does not support Group Policy Management Console (GPMC) on 64-bit platforms of Windows; thus, One Identity does not support managing group policies through the Control Center on Windows 2003 64-bit and Windows 2003 R2 64-bit, XP 64-bit platforms. See *Group Policy Management Console with Service Pack 1* for more information.

To launch the Group Policy Management Console

1. From the **Group Policy** window, click **Launch GPMC** from the **Actions** menu.

Tools

The **Tools** link on the Control Center gives you access to:

- **Authentication Services**

  Direct links to installed applications and tools related to Authentication Services.

- **Additional One Identity Products**

  Direct links to other One Identity product plugins.

  NOTE: The **Additional One Identity Products** link is only available if you have installed other One Identity products such as Defender, Authentication Services for Smart Cards, or One Identity Active Roles.

- **Other Tools**

  Direct links to tools related to Authentication Services.

  NOTE: The **Other Tools** link is only available if you have installed the Group Policy Management Console.

- **Documentation**

  Direct links to Authentication Services documentation.

Preferences

Authentication Services stores certain preferences and settings in Active Directory. This information is used by Authentication Services clients and management tools so that behavior remains consistent across all platforms and tools. The **Preferences** window allows you to configure these settings and preferences:

- Licensing
- Global Unix Options
- Logging Options
- Custom Unix Attributes

Licensing

The **Licensing** section of the **Preferences** window in the Control Center displays a list of installed license files. You can add and remove license files at any time. The license files are stored in Active Directory and Authentication Services Unix hosts automatically download and apply new license files from Active Directory.

Refer to Licensing Authentication Services on page 10 for more information about licensing requirements.

Adding licenses using the Control Center

To add licenses using the Control Center

1. Click the **Preferences** navigation button on the left panel of the Control Center.
2. Expand the **Licensing** section.

   The list box displays all licenses currently installed in Active Directory.
3. Click **Add a license** from the **Actions** menu.
4. Browse for the license file and click **Open**.

   The license appears in the list box.

NOTE: Unix hosts check for new licenses when the host is joined to the domain or every 24 hours by default. This can be changed by modifying the `configuration-refresh-interval` setting in `vas.conf`.

5. To remove a license, select it and click **Remove license**.
6. To restore a removed license, click **Undo Remove**.

Global Unix Options

The **Global Unix Options** section displays the currently configured options for Unix-enabling users and groups.

Click **Modify Global Unix Options** to change these settings.

NOTE: Authentication Services uses the **Global Unix Options** when enabling users and groups for Unix login.

Table 17: Unix user defaults

Option	Description
Require unique user login names	Select to require a unique user login name attribute within the forest.
Require unique UID on users	Select to require a unique user's Unix ID (UID) number within the forest.
Minimum UID Number	Enter a minimum value for the Unix User ID (UID) number. Typically, you set this to a value higher than the highest UID among local Unix users to avoid conflicts with users in Active Directory and local user accounts.
Maximum UID Number	Enter a maximum value for the Unix User ID (UID) number. Typically, you would not change this value unless you have a legacy Unix platform that does not support the full 32-bit integer range for UID number.
Primary GID Number	Enter the default value for the Primary GID number when Unix-enabling a user.
Set primary GID to UID	Select to set the primary GID number to the User ID number.
Default Comments (GECOS)	Enter any text in this box.
Login Shell	Enter the default value for the login shell used when Unix-enabling a user.
Home Directory	Enter the default prefix used when generating the home directory

Option	Description
	attribute when Unix-enabling a user. The default value is `/home/`; use a different value if your Unix user home directories are stored in another location on the file system. Authentication Services uses the user's effective Unix name when generating the full home directory path.
Use lowercase user name for home directory	Select to use a lower-case representation of the user's effective Unix name when generating the full home directory path as a user is Unix-enabled.

Table 18: Unix group defaults

Option	Description
Require unique Group Names	Select to require a unique Unix group name attribute within the forest.
Require unique GID Number	Select to require a unique Unix Group ID (GID) attribute within the forest.
Minimum GID Number	Enter the minimum value for the Unix Group ID (GID). Typically, this is set to a value higher than the highest GID among local Unix groups to avoid conflicts with groups in Active Directory and local group accounts.
Maximum GID Number	Enter the maximum value for the Unix Group ID (GID). Typically, you would not change this value unless you have a legacy Unix platform that does not support the full 32-bit integer range for GID.

These options control the algorithms used to generate unique user and group IDs.

Table 19: Unique IDs

Option	Description
Object GUID Hash	An ID generated from a hash of the user or group object GUID attribute. This is a fast way to generate an ID that is usually unique. If the generated value conflicts with an existing value, the ID is re-generated by searching the forest.
Samba Algorithm	An ID generated from the SID of the domain and the RID of the user or group object. This method works well when there are few domains in the forest. If the generated value conflicts with an existing value, the ID is re-generated by searching the forest.

Option	Description
Legacy Search Algorithm	An ID generated by searching for existing ID values in the forest. This method generates an ID that is not currently in use.

Modifications you make to these **Global Unix Options** take effect after you restart the Microsoft Management Console (MMC).

BEST PRACTICE: It is a best practice to either use the generated default IDs or set the ID manually. Mixing the two methods can lead to ID conflicts.

Logging Options

The **Logging Options** section allows you to enable logging for all Authentication Services Windows components. This setting only applies to the local computer. Logging can be helpful when trying to troubleshoot a particular problem. Because logging causes components to run slower and use more disk space, you should set the **Log Level** to **Disabled** when you are finished troubleshooting.

Enabling debug logging on Windows

To enable debug logging for all Authentication Services Windows components

1. Open Control Center and click the **Preferences** navigation button on the left panel.
2. Expand the **Logging Options** section.
3. Open the **Log level** drop-down menu and set the log level to **Debug**.

   Debug generates the most log output. Higher levels generate less output. You can set the **Log level** to **Disabled** to disable logging.
4. Click  to specify a folder location where you want to write the log files.

   Authentication Services Windows components log information into the specified log folder the next time they are loaded. Each component logs to a text file named after the DLL or EXE that generates the log message.

Custom Unix Attributes

The Unix schema attributes are fully customizable in Authentication Services. The **Custom Unix Attributes** section allows you to see which LDAP attributes are mapped to Unix attributes. You can modify this mapping to enable Authentication Services to work with any schema configuration. To customize the mapping, you select a schema template or specify your own custom attributes. A schema template is a pre-defined set of common mappings which adhere to common schema extensions for storing Unix data in Active Directory. Authentication Services supports the following schema templates if the required schema is installed:

Table 20: Unix schema attributes

Schema Template	Description
Schemaless	A template that encodes Unix attribute data in an existing multi-valued attribute.
Windows R2	A template that uses attributes from the Windows 2003 R2 schema extension.
Services for Unix 2.0	A template that uses attributes from the SFU 2.0 schema extension.
Services for Unix 3.0	A template that uses attributes from the SFU 3.0 schema extension.

BEST PRACTICE: Use a schema designed for storing Unix data in Active Directory whenever possible. Schemas designed for storing Unix data in Active Directory include: Windows 2003 R2, SFU 2, and SFU 3. Only use "schemaless" or custom mappings if it is impossible to make schema extensions in your environment.

NOTE: If you are running Authentication Services without an application configuration in your forest and your domain supports Windows 2003 R2, you can enable Authentication Services to use the Windows 2003 R2 schema. However, note that some functionality provided by the Authentication Services application configuration will be unavailable. For more information, see *Configure Windows 2003 R2 Schema* in the mangement console online help.

Active Directory schema extensions

Authentication Services stores Unix identity and login information in Active Directory. One Identity designed Authentication Services to provide support for the following standard Active Directory schema extensions.

Table 21: Active Directory schema extensions

Schema extension	Description
Windows 2003 R2 Schema	This schema extension is provided by Microsoft and adds support for the PosixAccount auxiliary class, used to store Unix attributes on user and group objects.
Services for Unix 2.0	Microsoft provides this schema extension with the Services for Unix 2.0 set of tools. It adds custom attributes to user and group objects, used to store Unix account information.
Services for Unix 3.0	Microsoft provides this schema extension with the Services for Unix 3.0 set of tools. It adds custom attributes to user and group objects, used to store Unix account information.

It is possible to customize the schema setup to work with any schema configuration with Authentication Services. No schema extensions are necessary with the new "schemaless" storage feature. When you configure Authentication Services for the first time, Authentication Services attempts to auto-detect the best schema configuration for your environment. The schema configuration is a global application setting that applies to all Authentication Services management tools and Unix agents. You can change the detected settings at any time using Control Center.

Configuring a custom schema mapping

If you do not have a schema that supports Unix data storage in Active Directory, you can configure Authentication Services to use existing, unused attributes of users and groups to store Unix information in Active Directory.

To configure a custom schema mapping

1. Open the Control Center and click the **Preferences** on the left navigation panel.
2. Expand the **Custom Unix Attributes** and click **Customize**.
3. Type the LDAP display names of the attributes that you want to use for Unix data. All attributes must be string-type attributes except **User ID Number**, **User Primary Group ID**, and **Group ID Number**, which may be integers. If an attribute does not exist or is of the wrong type, the border will turn red indicating that the LDAP attribute is invalid.

   NOTE: When customizing the schema mapping, ensure that the attributes used for **User ID Number** and **Group ID Number** are indexed and replicated to the global catalog.

   For more information, see Active Directory optimization on page 77.
4. Click **OK** to validate and save the specified mappings in Active Directory.

Active Directory optimization

Indexing certain attributes used by the Authentication Services Unix agent can have a dramatic effect on the performance and scalability of your Unix and Active Directory integration project. The **Custom Unix Attributes** panel in the **Preferences** section of Control Center displays a warning if the Active Directory configuration is not optimized according to best practices.

One Identity recommends that you index the following attributes in Active Directory:

- User UID Number
- User Unix Name
- Group GID Number
- Group Unix Name

NOTE: LDAP display names vary depending on your Unix attribute mappings.

It is also a best practice to add all Unix identity attributes to the global catalog. This reduces the number of Active Directory lookups that need to be performed by Authentication Services Unix agents.

Click the **Optimize Schema** link to run a script that updates these attributes as necessary.

NOTE: The **Optimize Schema** option is only available if you have not optimized the Unix schema attributes defined for use in Active Directory.

This operation requires administrative rights in Active Directory. If you do not have the necessary rights to optimize your schema, it generates a schema optimization script. You can send the script to an Active Directory administrator who has rights to make the necessary changes.

All schema optimizations are reversible and no schema extensions are applied in the process.

Learning the basics

The topics in this section help you learn how to do some basic system administration tasks using the Control Center and Management Console for Unix.

NOTE: The exercises in this section assume that you have successfully installed Authentication Services and Management Console for Unix and have added a host to the console and joined it to Active Directory. See Prepare Unix hosts on page 54.

This section shows you how to create the following test user and group accounts used in various examples:

- A local group name called **localgroup**
- A local user object called **localuser**
- An Active Directory group object called **UNIXusers**
- An Active Directory user object called **ADuser**

One Identity recommends that you work through the topics in this section in order as a self-directed "test drive" of some of the key product features. You will learn how easy it is to manage your users and groups from the mangement console.

Adding a local group

You can use the mangement console to remotely add a local group to the host.

NOTE: This topic instructs you to set up a local group by the name of "localgroup" referred to by other examples in this guide.

To add a local group to the host

1. From the Management Console for Unix **Hosts** | **All Hosts** view, double-click a host name to open its properties.
2. Select the **Groups** tab and click **Add Group**.
3. In the **Add New Group** dialog, enter **localgroup** as a local group name in the **Group Name** box and click **Add Group**.
4. In the **Log on to Host** dialog, enter your credentials and click **OK**.

   NOTE: This task requires elevated credentials. Credential information is entered by default from the cache.

   The new local group account is added to the system and mangement console.

Adding a local user account

NOTE: This topic instructs you to set up a local user by the name of "localuser" referred to by other examples in this guide.

To add a local user account

1. From the **All Hosts** view, double-click a host name to open its properties.
2. Select the **Users** tab from the host properties and click **Add User**.
3. In the **Add New User** dialog:
   a. Enter **localuser** as a new local user name in the **Name** box.
   b. Click **Select Group** browse button next to the **GID** box, to find and select the **local group** account you set up in Adding a local group on page 78.

      You can also the navigation buttons at the bottom of the list to find and select a group.
   c. Click the **Select Shell** browse button to find and select a local login shell.
   d. Enter and re-enter a password of your choice and click **Add User** to add this new local user.
4. In the **Log on to Host** dialog, enter your credentials to log in to the host and click **OK**.

   NOTE: This task requires elevated credentials. The mangement console enters this information by default from the cache.

   The new local user account is added to the system and mangement console.

At this point the new local user is valid for local authentication with the password you just set.

Adding an Active Directory group account

Authentication Services provides additional tools to help you manage different aspects of migrating Unix hosts into an Active Directory environment. Links to these tools are available from **Tools** in the Control Center.

NOTE: This topic instructs you to set up an Active Directory group by the name of "UNIXusers" referred to by other examples in this guide.

To create a new group in Active Directory

1. In the Control Center, navigate to **Tools** and click the link for **Authentication Services Extensions for Active Directory Users and Computers**.

   The **Active Directory Users and Computers** Console opens.

   NOTE:

   - Windows Vista/Windows 7: You must have the Remote Server Administration Tools installed and enabled.
   - Windows2003/Windows XP: You must have the Windows 2003 Server Administration Tools installed.

2. Expand the **domain** folder and select the **Users** folder.
3. Click the **New Group** icon button.

   The **New Object - Group** dialog opens.
4. Enter **UNIXusers** in the **Group name** box and click **OK**.

Adding an Active Directory user account

NOTE: The following procedure instructs you to use ADUC (Active Directory Users and Computers) to set up an Active Directory user by the name of "ADuser" referred to by other examples in this guide.

To create an Active Directory user account

1. In the **Active Directory Users and Computers** console, select the **Users** folder and click the **New User** icon button.
2. On the **New Object - User** dialog, enter information to define a new user named **ADuser** and click **Next**.

   The **New Object - User** wizard guides you through the user setup process.
3. When you enter a password, clear the **User must change password at next logon** option, before you click **Next**.
4. Click **Finish**.

5. Close **Active Directory Users and Computers** and return to the mangement console.

Changing the default Unix attributes

You can modify the Unix attributes that are generated by default when users are Unix-enabled. To change the Login Shell you must have rights to create and delete child objects in the Authentication Services application configuration in Active Directory.

To change the default Unix attributes

1. Click the **Preferences** navigation button on the left panel of the Control Center.
2. Expand **Global Unix Options**.

   The window displays the current settings for Unix-enabling users, groups and the method used for creating unique IDs.
3. Click **Modify Global Unix Options** on the right side of the window.

   The **Modify Global Options** dialog opens.
4. Change the **Login Shell** to **/bin/bash** and click **OK**.

   The defaults are saved to Active Directory.

NOTE: Now, when you Unix-enable a user from Active Directory Users and Computers, PowerShell, or the Unix command line, the login shell defaults to `/bin/bash`. You can customize the other Unix defaults similarly.

Active Directory account administration

The topics that follow show you how to perform Active Directory account administration from Management Console for Unix for hosts that are joined to Active Directory.

Enabling local user for AD authentication

This feature, also known as user mapping, allows you to associate an Active Directory user account with a local Unix user. Allowing a local user to log in to a Unix host using Active Directory credentials enables that user to take advantage of the benefits of Active Directory security and access control.

To enable a local user for Active Directory authentication

1. From the mangement console **Hosts** | **All Hosts** view, double-click a host to open its properties.

2. Select the **Users** tab and double-click the **localuser** account to open its properties.

   NOTE: To set up this local user account, see Adding a local user account on page 79.

3. In the **AD Logon** tab, select the **Require an AD Password to logon to Host** option, and click **Select**.
4. In the **Select AD User** dialog, click the **Search** button to populate the list of Active Directory users, select the **ADuser** account, and click **OK**.

   NOTE: To set up this Active Directory user, see Adding an Active Directory user account on page 80.

5. On the localuser's properties, click **OK**.
6. In the **Log on to Host** dialog, verify your credentials to log in to the host and click **OK**.

   You have now mapped a local user to an Active Directory user and the mangement console indicates that the local user account requires an Active Directory password to log onto the Host in the **AD User** column.

You can also map multiple Unix users to use a single Active Directory account using the **Require AD Logon** pane on the **All Local Users** tab.

To assign (or "map") a Unix user to an Active Directory user

1. From the **All Local Users** tab, select one or more local Unix users.
2. In the **Require AD Logon** pane, click the **Search** button to populate the list of Active Directory users.

   (Click the **Directory** button to search in a specific folder.)

3. Select an Active Directory user and click the **Require AD Logon to Host** button at the bottom of the **Require AD Logon** pane.
4. In the **Log on to Host** dialog, verify your credentials to log in to the host and click **OK**.

   NOTE: This task requires elevated credentials.

The Active Directory user assigned to the selected local Unix users displays in the **AD User** column of the **All Local Users** tab.

Testing the mapped user login

Once you have mapped a local user to an Active Directory user, you can log in to the local Unix host using your local user name and the Active Directory password of the Active Directory user to whom you are mapped.

To test the mapped user login

1. From the Control Center, under **Login to remote host**, enter:
   - **Home name**: The Unix host name.
   - **User name**: The local user name, **localuser**.

   Click **Login** to log in to the Unix host with your local user account.
2. If the **PuTTY Security Alert** dialog opens, click **Yes** to accept the new key.
3. Enter the password for **ADuser**, the Active Directory user account you mapped to **localuser**, when you selected the **Require an AD Password to logon to Host** option on the user's properties.
4. At the command line prompt, enter `id` to view the Unix account information.
5. Enter `/opt/quest/bin/vastool klist` to see the credentials of the Active Directory user account.
6. Enter `exit` to close the command shell.

You just learned how to manage local users and groups from Management Console for Unix by mapping a local user account to an Active Directory user account. You tested this by logging into the Unix host with your local user name and the password for the Active Directory user account to whom you are mapped.

Unix-enabling an Active Directory group

To Unix-enable an Active Directory group

1. On the mangement console's **Active Directory** tab, open the **Find** box drop-down menu and choose **Groups**.
2. Enter a group name, such as **UNIX**, in the **Search by name** box and press **Enter**.
3. Double-click the group name, such as **UNIXusers**, to open its properties.

   NOTE: To set up this Active Directory user account, see Adding an Active Directory group account on page 80.
4. On the **Unix Account** tab, select the **Unix-enabled** option and click **OK**.

Unix-enabling an Active Directory user

To Unix-enable an Active Directory user

1. On the mangement console's **Active Directory** tab, open the **Find** box drop-down menu and choose **Users**.
2. Click next to the **Search by name** box to search for all Active Directory users. Or, enter a portion of your **ADuser** logon name in the **Search by name** box and press **Enter**.

3. Double-click **ADuser**, the Active Directory user name, to open its properties.
4. On the **Unix Account** tab, select the **Unix-enabled** option.

   It populates the properties with default Unix attribute values.
5. Make other modifications to these settings, if necessary, and click **OK** to Unix-enable the user.

   NOTE: There are additional settings that you can set using PowerShell which allows you to validate entries for the GECOS, Home Directory, and Login Shell attributes. Refer to Use Authentication Services PowerShell on page 98 to learn more about that.

   Once enabled for Unix, you can log on to the host with that Active Directory user's log on name and password.

Testing the Active Directory user login

Now that you have Unix-enabled an Active Directory user, you can log in to a local Unix host using your Active Directory user name and password.

To test the Active Directory login

1. From the Control Center, under **Login to remote host**, enter:
   - **Host name**: Tthe Unix host name.
   - **User name**: The Active Directory user name, such as **ADuser**.

   Click **Login** to log in to the Unix host with your Active Directory user account.
2. Enter the password for the Active Directory user account.
3. At the command line prompt, enter `id` to view the Unix account information.
4. After a successful log in, verify that the user obtained a Kerberos ticket by entering:

   ```
   /opt/quest/bin/vastool klist
   ```

   The `vastool klist` command lists the Kerberos tickets stored in a user's credentials cache. This proves the local user is using the Active Directory user credentials.
5. Enter `exit` to close the command shell.

You just learned how to manage Active Directory users and groups from Management Console for Unix by Unix-enabling an Active Directory group and user account. You tested this out by logging into the Unix host with your Active Directory user name and password. Optionally, you can expand on this tutorial by creating and Unix enabling additional Active Directory users and groups and by testing different Active Directory settings such as account disabled and password expired.

Running reports

You can run various reports that capture key information about the Unix hosts you manage from the mangement console and the Active Directory domains joined to these hosts from the **Reports** view on the **Reporting** tab.

NOTE: The Active Directory reports are only available when you are logged on as an Active Directory account in the **Manage Hosts** role.

To run reports

1. Ensure the hosts for which you want to create reports have been recently profiled.

   Reports only generate data gathered from the clients during a profile procedure. Profiling imports information about the host, including local users and groups.

   NOTE: You can configure the mangement console to profile hosts automatically. For more information, see Configuring automatic profiling on page 57.

2. From the mangement console, click the **Reporting** tab.
3. From the **Reports** view, expand the report group names to view the available reports, if necessary.
   - **Host Reports**

     Unix host information gathered during the profiling process
   - **User Reports**

     Local and Active Directory user information
   - **Group Reports**

     Local and Active Directory group information
   - **Access & Privileges Reports**

     User access information
   - **License Usage Reports**

     Product licensing information.
4. Use one of the following methods to select a report:
   - Double-click a report name in the list (such as the **Unix Host Profiles** report).
   - Right-click a report name and select **Run report**.
   - Click the report icon next to a report.

   The selected report name opens a new tab on the **Reports** view that describes the report and provides some report parameters you can select or clear to add or exclude details on the report.
5. Optionally clear parameters to exclude information from the report.

6. To create a report, either:
    - Click **Preview** to see a sample of the report in a browser.
    - Open the **Export** drop-down menu and select the format you want to use for the report: **PDF** or **CSV** (if available).

    NOTE: If the CSV report does not open, you may need to reset your internet options. See *CSV or PDF Reports Do Not Open* in the online help for details.

    By default, the mangement console creates reports in the application data directory:

    - On Windows XP/2003 Server:

      ```
      %SystemDrive%:\Documents and Settings\All Users\Application Data\Quest Software\Management Console for Unix\reports
      ```

    - On Windows 2008 Server/Vista/7:

      ```
      %SystemDrive%:\ProgramData\Quest Software\Management Console for Unix\reports
      ```

    - On Unix/Mac OS X:

      ```
      /var/opt/quest/mcu/reports
      ```

    NOTE: You may need to reconfigure your browser preferences to allow you to save the report in a specific folder.

    It launches a new browser or application page and displays the report in the selected format.

NOTE: When generating multiple reports simultaneously or generating a single report that contains a large amount of data, One Identity recommends that you increase the JVM memory. See *Tune JVM Memory* in the online help for details.

Reports

The mangement console provides comprehensive reporting which includes reports that can help you plan your deployment, consolidate Unix identity, secure your hosts and troubleshoot your identity infrastructure. The following tables list the reports that are available in Management Console for Unix.

NOTE: Report availability depends on several factors:

- **User Log-on Credentials**: While some reports are available when you are logged in as **supervisor**, there are some reports that are only available when you are logged on as an Active Directory user. See *Active Directory Configuration* in the online help for details.
- **Roles and Permissions**: Reports are hidden if they are not applicable to the user's console role. See *Console Roles and Permissions System Settings* in the online help for details. For example, you must have an activated policy server to activate the sudo-related reports.

Host reports

The following reports provide Unix host information that is gathered during the profiling process.

Table 22: Host reports

Report	Description
Authentication Services Readiness	Provides a snapshot of the readiness of each host to join Active Directory. This report is best used for planning and monitoring migration projects. The basic report includes the following information: • Total number of hosts • Total number, percentage, and names of the hosts ready to join • Total number, percentage, and names of the hosts ready to join with advisories • Total number, percentage, and names of the hosts not ready to join • Total number of hosts not checked for AD readiness Use the following report parameters to define details to include in the report. • Joined to AD • Ready to Join AD • Ready to Join AD with Warnings • Not Ready to Join AD • Not Checked for Readiness NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Hosts** role.
Privilege Manager Readiness	Provides a snapshot of the readiness of each host to join a policy group. The basic report includes the following information: • Total number of hosts • Total number, percentage, and names of the hosts ready to join • Total number, percentage, and names of the hosts not ready to join • Total number of hosts not checked for readiness Use the following report parameters to define details to include in the report. • Joined to a policy group • Ready to join a policy group

Report	Description
	• Ready to join a policy group with warnings • Not ready to join a policy group • Not checked for readiness NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Sudo Policy** role or the **Audit Sudo Policy** role.
Unix Computers in AD	Lists all Unix computers in Active Directory in the requested scope. By default, this report is created using the default domain as the base container. Browse to search Active Directory to locate and select a different base container to begin the search. NOTE: This report is available when you are logged on as an Active Directory account in the **Manage Hosts** role.
Unix Host Profiles	Summarizes information gathered during the profiling process of each managed host. This report includes the following information: • Total number of hosts included in the report • Host Name, IP Address, OS, Hardware • Sudo version number Use the following report parameters to define details to include for each host. • Authentication Services Properties • Privilege Manager Properties • Local Users • Local Groups • Host SSH Keys • Installed One Identity Software NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Hosts** role.

User reports

The following reports provide local and Active Directory user information.

Table 23: User reports

Report	Description
AD User Conflicts	Returns all users with Unix User ID numbers (UID numbers) assigned to other Unix-enabled user accounts. By default, it creates this report using the default domain as the base container. Browse to search Active Directory to locate and select a different base container to begin the search. NOTE: This report is available when you are logged on as an Active Directory account in the **Manage Hosts** role.
Local Unix User Conflicts	Identifies local user accounts that would conflict with a specified user name and UID on other hosts. You can use this report for planning user consolidation across your hosts. This report includes the following information: • Host Name, DNS Name, or IP Address where a conflict would occur • User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory, and Login Shell for each host where conflicts exist Use the following report parameters to define the user name and UID number that would cause a conflict with existing local user accounts: • User Name is • UID Number is NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the *Manage Hosts* role.
Local Unix Users	Lists all users on all hosts or lists the hosts where a specific user account exists in `/etc/passwd`. This report includes the following information: • Host Name, DNS Name, or IP Address where the user exists • User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory, and Login Shell for each host where the user exists If you do not define a specific user, it includes all local users on each profiled host in the report. To locate a specific user, use the following report parameters: • User Name contains • UID Number is • Primary GID Number is • Comment (GECOS) contains • Home Directory contains

Report	Description
	• Login Shell contains NOTE: When you specify multiple report parameters, it uses the AND expression; therefore, ALL of the selected parameters must be met in order to locate the user account. NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Hosts** role.
Local Unix Users with AD Logon	Identifies the local user accounts that are required to use Active Directory credentials to log onto the Unix hosts. This report includes the following information for hosts that are joined to an Active Directory domain: • Host Name, DNS Name, or IP Address of hosts where users exist that are required to log on using their AD credentials • User Name, UID Number, Primary GID Number, and Comment (GECOS) of local user account • The SAM account Name of the Active Directory account that the local user account must use to log on NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent. NOTE: This report is only available when the host has Authentication Services 4.*x* or later installed and is joined to Active Directory. You must be logged in with an Active Directory account in the **Manage Hosts** role.
Master /etc/-passwd List	Provides a consolidated list of all user accounts from all hosts, excluding any local users marked as system users. This report includes the following information: • Username • Empty password • UID • GID • GECOS • Home directory path • Account's shell You can consolidate the list of user accounts by matching values for accounts across multiple hosts. Accounts found with matching values are listed as a single local account. This list is best used for migrating local

Report	Description
	users to Active Directory. Indicate how you want to match user accounts by selecting the value parameters that you want to match: • Username • UID • GID • GECOS • Home Directory • Shell Optionally, you can include the host name for the accounts, as well: • Include the host name for accounts NOTE: If you select the **Include the host name for accounts** option, the mangement console adds a column to the `Master_etc_passwdList .csv` file to identify the host for each user account. One Identity provides the **Host** column information to help you resolve the entries in the file. However, before you import the `.cvs` file into the Unix Account Import Wizard, you must remove the **Host** column. You can easily migrate local users to Active Directory by exporting the `Master /etc/passwd List` report, then importing it into the Unix Account Import Wizard, accessible from the Authentication ServicesControl Center's **Tools** link. The **Unix Account Import** wizard is a versatile tool that helps migrate Unix account information to Active Directory. It is especially well-suited to small, one-shot import tasks such as importing all the local user accounts from a specific Unix host. The Unix Account Import Wizard can import Unix data as new user and group objects or use the data to Unix-enable existing users and groups. NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Hosts** role.
Unix-Enabled AD Users	Lists all Active Directory users that have Unix user attributes.

Report	Description
	NOTE: • A User object is considered to be 'Unix-enabled' if it has values for the UID Number, Primary GID Number, Home Directory, and Login Shell. • If Login Shell is `/bin/false`, the user is considered to be disabled for Unix or Linux logon. • Account Disabled indicates whether the Active Directory User account is enabled or disabled. By default, it creates this report using the default domain as the base container. Browse to search Active Directory to locate and select a different base container to begin the search.
	NOTE: This report is only available if you have configured the mangement console to recognize Active Directory objects (see *Configuring the Console to Recognize Unix Attributes in AD* in the online help), and you are logged on as an Active Directory account in the **Manage Hosts** role.

Group reports

The following reports provide local and Active Directory group information.

Table 24: Group reports

Report	Description
AD Group Conflicts	Lists all Active Directory groups with Unix Group ID (GID) numbers assigned to other Unix-enabled groups. By default, it creates this report using the default domain as the base container. Browse to search Active Directory to locate and select the base container to begin the search. NOTE: This report is available when you are logged on as an Active Directory account in the **Manage Hosts** role.
Local Unix Groups	Identifies the hosts where a specific group exists in `/etc/group`. This report includes the following information: • Host Name, DNS Name, or IP Address where the group exists • Group Name, GID Number, and members for each host where the group exists If you do not specify a group, it includes all local groups on each profiled host in the report.

Report	Description
	To locate a specific group, use the following report parameters: • **Group Name contains** • **GID Number is** • **Member contains** • **Include all group members in report** NOTE: The **Member contains** field accepts multiple entries separated by a comma. Spaces are taken literally in the search. For example, entering: • **adm, user** searches for members whose name contains "adm" or "user" • **adm,user** searches for members whose name contains "adm" or "user" NOTE: When you specify multiple report parameters (for example, **Group Name contains**, **GID Number is**, and **Member contains**), it uses the AND expression; therefore, ALL of the selected parameters must be met in order to locate a group. In addition, it includes all of the group members in the report by default, but you can clear the **Include all group members in report** option. NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Hosts** role.
Unix-Enabled AD Groups	Lists all Active Directory groups that have Unix group attributes. NOTE: A Group object is considered 'Unix-enabled' if it has a value for the GID Number. By default, it creates this report using the default domain as the base container. Browse to search Active Directory to locate and select a different base container to begin the search. NOTE: This report is only available if you have configured the mangement console to recognize Active Directory objects (see *Configuring the Console to Recognize Unix Attributes in AD* in the online help), and you are logged on as an Active Directory account in the *Manage Hosts* role.

Access & Privileges reports

The following reports provide user access information.

NOTE: The Access & Privileges reports do not report on users and groups from a NIS domain.

Table 25: Access & Privileges reports

Report	Description
Access & Privileges by Host	Identifies all users with log-on access to hosts and the commands the users can run on the hosts. This report includes the following information: • Total number of users that can log on to the host • The users that can log on to the host • The commands users can run on the host • The runas aliases for which the user can run commands on the host • The commands the runas alias can run on the host **Browse** to select a host. Optionally, select the **Show detailed report** option. NOTE: This report is available when you are logged on as the **supervisor** or as an Active Directory account in the **Manage Sudo Policy**, **Manage PM Policy**, **Audit Sudo Policy**, or **Audit PM Policy** roles. You must have an active policy group for Privilege Manager to run this report; you can only include hosts that are joined to a policy group.
Access & Privileges by User	Identifies the users with logon access to hosts, the commands that user can run on each host, and the "runas aliases" information for that user. This report includes the following information: • Total number of hosts where the user can log on • The hosts where the user can log on • The commands the user can run on each host • The runas aliases for which the user can run commands on each host • The commands the runas alias can run on each host Use the following report parameters to specify the user to include in the report: • A local user (default) • An AD user **Browse** to select a user. Optionally select the **Show detailed report** option.

Report	Description
	NOTE: This report is available when you are logged on as the **supervisor** or as an Active Directory account in the **Manage Sudo Policy**, **Manage PM Policy**, **Audit Sudo Policy**, or **Audit PM Policy** roles. You must have an active policy group for Privilege Manager to run this report; you can only include hosts that are joined to a policy group.
Commands Executed	Provides details about the commands executed by users on hosts joined to a policy group, based on their privileges and recorded as events or captured in keystroke logs by Privilege Manager. This report allows you to search for commands that have been recorded as part of events or keystroke logs for a policy group and includes the following information: • Command name • User who executed the command • Date and time the command was executed • Host where the command was executed Use the following report parameters to define details in the report: • Policy Group • Command • Host • Log status • Date NOTE: You can use wildcards in the text string you enter in the **Command** box, such as * and ?. NOTE: This report is available when you are logged on as the **supervisor** or as an Active Directory account in the **Manage Sudo Policy**, **Manage PM Policy**, **Audit Sudo Policy**, or **Audit PM Policy** roles. You must have an active policy group for Privilege Manager to run this report; you can only include hosts that are joined to a policy group.
Console Access and Permissions	Lists users who have access to the mangement console based on membership in a console role and the permissions assigned to that role. This report includes the following information: • List of roles • List of permissions assigned to each role • List and number of members assigned to each role

Report	Description
	NOTE: This report is available when you are logged on as the **supervisor** or an Active Directory account in the **Manage Console Access** role. However, when you access this report as **supervisor**, the mangement consolerequires that you authenticate to Active Directory.
Logon Policy for AD User	Identifies the hosts where Active Directory users have been granted logon permission. This report includes the following information for hosts joined to an Active Directory domain: • Total number of hosts where the AD user has access • List of hosts where the AD user has access Specify the Active Directory users to include in the report: • All AD users (default) • Select AD user **Browse** to search Active Directory to locate and select an Active Directory user. NOTE: The report may show both the Active Directory login name and local user names in the **Login Name** column for a selected AD user account because an Active Directory user account can have one or more local user accounts mapped to it. NOTE: Only hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent are included in this report. NOTE: This report is available when you are logged on as an Active Directory account in the **Manage Hosts** role.
Logon Policy for Unix Host	Identifies the Active Directory users that have been explicitly granted log-on permissions for one or more Unix computers. This report includes the following information for hosts joined to an Active Directory domain: • Host Name, DNS Name, or IP Address of the host selected for the report • Users that have been granted permission to log on Specify the managed hosts to include in the report: • All profiled hosts (default) • Select host **Browse** to locate and select a managed host that is joined to Active Directory.

Report	Description
	NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent. NOTE: This report is available when you are logged on as an Active Directory account in the **Manage Hosts** role.
Policy Changes	Provides details of changes made to a policy for a Privilege Manager policy group. This report includes the following information: • Name of the user that made changes to the policy • Version number for the changes • Time and date the changes were saved and actively used to enforce policy • Changes made to the policy based on version Select a policy group. Select to: • Show all changes to the policy • Show only changes for a specific pmpolicy file (not available for sudo-based policy) • Show changes to the policy for changes for one or more revisions NOTE: This report is available when you are logged on as the **supervisor** or as an Active Directory account in the **Manage Sudo Policy**, **Manage PM Policy**, **Audit Sudo Policy**, or **Audit PM Policy** roles. You must have an active policy group for Privilege Manager to run this report; you can only include hosts that are joined to a policy group.

Product licenses usage report

The following report provides product licensing information.

Table 26: Product licenses usage reports

Report	Description
Product License Usage	Provides a summary of all licensing information. This report includes the following information for hosts managed by the console: • Product • Purchased licenses • Used licenses

Use Authentication Services PowerShell

Authentication Services includes PowerShell modules that provide a "scriptable" interface to many Authentication Services management tasks. You can access a customized PowerShell console from the Control Center **Tools** navigation link.

You can perform the following tasks using PowerShell cmdlets:

- Unix-enable Active Directory users and groups
- Unix-disable Active Directory users and groups
- Manage Unix attributes on Active Directory users and groups
- Search for and report on Unix-enabled users and groups in Active Directory
- Install product license files
- Manage Authentication Services global configuration settings
- Find Group Policy objects with Unix/Mac OS X settings configured

Using the Authentication Services PowerShell modules, it is possible to script the import of Unix account information into Active Directory.

Unix-enabling a user and user group (PowerShell Console)

The following procedure explains how to Unix-enable a user and user group using the Authentication Services PowerShell Console.

To Unix-enable a user and user group

1. From the Control Center, navigate to **Tools | Authentication Services**.
2. Click **Authentication Services PowerShell Console**.

   NOTE: The first time you launch the PowerShell Console, it asks you if you want to run software from this untrusted publisher. Enter `A` at the PowerShell prompt to import the digital certificate to your system as a trusted entity. Once you have done this, you will never be asked this question again on this machine.

3. At the PowerShell prompt, enter the following:

   ```
   Enable-QasUnixGroup UNIXusers | Set-QasUnixGroup -GidNumber 1234567
   ```

   NOTE: You created the `UNIXusers` group in a previous exercise. See Adding an Active Directory group account on page 80.

   Unix attributes are generated automatically based on the Default Unix Attributes settings that were configured earlier and look similar to the following:

```
ObjectClass             : group
DistinguishedName       : CN=UNIXusers,CN=Users,DC=example,DC=com
ObjectGuid              : 71aaa88-d164-43e4-a72a-459365e84a25
GroupName               : UNIXusers
UnixEnabled             : True
GidNumber               : 1234567
AdsPath                 : LDAP://windows.example.com/CN=UNIXusers,CN=Users,
                          DC=example,DC=com
CommonName              : UNIXusers
```

4. At the PowerShell prompt, to Unix-enable an Active Directory user using the default Unix attribute values, enter:

   ```
   Enable-QasUnixUser ADuser | Seet-QasUnixUser -PrimaryGidNumber 1234567
   ```

   The Unix properties of the user display:

   ```
   ObjectClass             : user
   DistinguishedName       : CN=ADuser,CN=Users,DC=example,DC=com
   ObjectGuid              : 5f83687c-e29d-448f-9795-54d272cf9f25
   UserName                : ADuser
   UnixEnabled             : True
   UidNumber               : 80791532
   PrimaryGidNumber        : 1234567
   Gecos                   :
   HomeDirectory           : /home/ADuser
   LoginShell              : /bin/sh
   AdsPath                 : LDAP://windows.example.com/CN=ADuser,CN=Users,
                             DC=example,DC=com
   CommonName              : ADuser
   ```

5. To disable the ADuser user for Unix login, at the PowerShell prompt enter:

   ```
   Disable-QasUnixUser ADuser
   ```

   NOTE: To clear all Unix attribute information, enter:

   ```
   Clear-QasUnixUser ADuser
   ```

   Now that you have Unix-disabled the user, that user can no longer log in to systems running the Authentication Services agent.

6. From the Control Center, under **Login to remote host**, enter:
   - **Host name**: The Unix host name.
   - **User name**: The Active Directory user name, **ADuser**.

   Click **Login** to log in to the Unix host with your Active Directory user account.

   A PuTTY window displays.

   NOTE: PuTTY attempts to log in using Kerberos, but will fail over to password authentication if Kerberos is not enabled or properly configured for the remote SSH service.

7. Enter the password for the Active Directory user account.

   You will receive a message that says `Access denied`.

PowerShell cmdlets

Authentication Services supports the flexible scripting capabilities of PowerShell to automate administrative, installation, and configuration tasks. A wide range of new PowerShell cmdlets are included in Authentication Services.

Table 27: PowerShell cmdlets

cmdlet name	Description
Add-QasLicense	Installs an Authentication Services license file in Active Directory. Licenses installed this way are downloaded by all Unix clients.
Clear-QasUnixGroup	Clears the Unix identity information from group object in Active Directory. The group is no longer Unix-enabled and will be removed from the cache on the Authentication Services Unix clients.
Clear-QasUnixUser	Clears the Unix identity information from a user object in Active Directory. The user is no longer Unix-enabled will be removed from the cache on the Authentication Services Unix clients.
Disable-QasUnixGroup	Unix-disables a group and will be removed from the cache on the Authentication Services Unix clients. Similar to Clear-QasUnixGroup except the Unix group name is retained.
Disable-QasUnixUser	Removes an Active Directory user's ability to log in on Unix hosts. (The user will still be cached on the Authentication Services Unix clients.)
Enable-QasUnixGroup	Enables an Active Directory group for Unix by giving a Unix GID number. The GID number is automatically generated.
Enable-QasUnixUser	Enables an Active Directory user for Unix. The required account attributes UID number, primary GID number, GECOS, login shell, and home directory are generated automatically.
Get-QasConfiguration	Returns an object representing the Authentication Services application configuration data stored in Active Directory.
Get-QasGpo	Returns a set of objects representing GPOs with Unix

cmdlet name	Description
	and/or Mac OS X settings configured. This cmdlet is in the `Quest.AuthenticationServices.GroupPolicy` module.
Get-QasLicense	Returns objects representing the Authentication Services product licenses stored in Active Directory.
Get-QasOption	Returns a set of configurable global options stored in Active Directory that affect the behavior of Authentication Services.
Get-QasSchema	Returns the currently configured schema definition from the Authentication Services application configuration.
Get-QasSchemaDefinition	Returns a set of schema templates that are supported by the current Active Directory forest.
Get-QasUnixGroup	Returns an object that represents an Active Directory group as a Unix group. The returned object can be piped into other cmdlets such as Clear-QasUnixGroup or Enable-QasUnixGroup.
Get-QasUnixUser	Returns an object that represents an Active Directory user as a Unix user. The returned object can be piped into other cmdlets such as Clear-QasUnixUser or Enable-QasUnixUser.
Get-QasVersion	Returns the version of Authentication Services currently installed on the local host.
Move-QasConfiguration	Moves the Authentication Services application configuration information from one container to another in Active Directory.
New-QasAdConnection	Creates an object that represents a connection to Active Directory using specified credentials. You can pass a connection object to most Authentication Services cmdlets to execute commands using different credentials.
New-QasArsConnection	Creates an object that represents a connection to an Active Roles Server using the specified credentials. You can pass a connection object to most Authentication Services cmdlets to execute commands using different credentials.
New-QasConfiguration	Creates a default Authentication Services application configuration in Active Directory and returns an object representing the newly created configuration.
Remove-QasConfiguration	Accepts a Authentication Services application configuration object as input and removes it from Active

cmdlet name	Description
	Directory. This cmdlet produces no output.
Remove-QasLicense	Accepts an Authentication Services product license object as input and removes the license from Active Directory. This cmdlet produces no output.
Set-QasOption	Accepts an Authentication Services options set as input and saves it to Active Directory.
Set-QasSchema	Accepts an Authentication Services schema template as input and saves it to Active Directory as the schema template that will be used by all Authentication Services Unix clients.
Set-QasUnixGroup	Accepts a Unix group object as input and saves it to Active Directory. You can also set specific attributes using command line options.
Set-QasUnixUser	Accepts a Unix user object as input and saves it to Active Directory. You can also set specific attributes using command line options.

Authentication Services PowerShell cmdlets are contained in PowerShell modules named `Quest.AuthenticationServices` and `Quest.AuthenticationServices.GroupPolicy`. Use the `Import-Module` command to import the Authentication Services commands into an existing PowerShell session.

Change Auditor for Authentication Services

Change Auditor for Authentication Services allows you to track changes and send alerts on:

- Changes to Active Directory objects and attributes
- Changes to Unix and Mac OS X settings in Group Policy Objects
- Changes to Product settings and configuration

Installing Change Auditor for Authentication Services

The following steps outline the basic procedure for installing Change Auditor for Authentication Services. See the *Change Auditor Installation Guide* to obtain detailed steps for installing Change Auditor for Authentication Services.

To install Change Auditor for Authentication Services

1. Insert the Authentication Services distribution media.

   The Autorun **Home** page displays.

   NOTE: If the Autorun **Home** page does not display, navigate to the root of the distribution media and double-click **autorun.exe**.

2. Click the **Setup** tab and select **Change Auditor for Authentication Services**.

   The **Change Auditor for Authentication Services for Active Directory** web page opens.

3. Click **Download** on the left navigation panel.
4. Follow the online instructions to gain access to the **Trial Download** page.
5. From the **Trial Download: Change Auditor for Active Directory** page, click the **Installation Guide** link.

One Identity Defender

One Identity Defender, another One Identity product, provides strong authentication functionality that makes it possible for an Active Directory user to use a hardware or software token to authenticate to Unix, Linux, or Mac OS X platforms.

Installing Defender

In order to use strong authentication, you must download and install Authentication Services Defender. See the *Defender Installation Guide* to obtain detailed steps for installing Authentication Services Defender.

NOTE: Defender installation requires a license file. A fully-functional 25-user license for it is included with Authentication Services.

The following steps outline the basic procedure for installing Defender. See the

To install Defender

1. Insert the Authentication Services distribution media.

   The Autorun **Home** page displays.

   NOTE: If the Autorun **Home** page does not display, navigate to the root of the distribution media and double-click **autorun.exe**.

2. From the **Home** page, click the **Setup** tab.
3. From the **Setup** tab, click **One Identity Defender**.

   The **One Identity Defender** web page opens.

4. Click the **Download** on the left navigation panel.

5. Follow the online instructions to gain access to the **Trial Download** page.
6. From the **Trial Download: Defender** page, click the **Defender Documentation Archive** link.
7. Once you have installed One Identity Defender, see the *One Identity Defender Integration Guide* located in the Control Center **Tools** page, or in the docs directory of the Authentication Services Installation media, for detailed configuration instructions about integrating Authentication Services Defender with Authentication Services.

11

Troubleshooting

This section lists some of the common installation problems that you may experience along with suggested resolutions.

Getting help from technical support

If you are unable to determine the solution to a problem, contact Technical Support for help.

NOTE: For more information, see About us on page 114.

Before you contact Support, please collect the following information:

1. Take a system information snapshot. To do this, run the following command as root:

   `/opt/quest/libexec/vas/scripts/vas_snapshot.sh`

   This produces an output file in `/tmp`.
2. Make note of the Unix attributes for the user that cannot log in (if applicable). To do this, capture the output from the following commands:

```
vastool -u host/ attrs <username>
id <username>
```

NOTE: Depending on your platform, you may need to run `id -a` instead of `id`.

3. Copy the text from any error messages that you see.
4. Save the results of running a "double su." To do this, log in as root and run `su <username>` note any error messages. Then run `su <username>` again and note any error messages.

Once you have collected the information listed above, contact Support at https://support.oneidentity.com/authentication-services/.

Disaster recovery

Since Authentication Services relies on Active Directory, follow Microsoft's best practices for keeping the database highly available. The Management Console for Unix and other administration tools, are not critical to the operation of Authentication Services and can quickly be reinstalled from scratch if needed.

Long startup delays on Windows

You may experience long delays (over a minute) when starting the Authentication Services Windows installer or certain Windows management tools such as Control Center. All Authentication Services Windows binaries are Authenticode-signed so that you can be sure that the binaries are authentic and have not been tampered with. This problem occurs when the .NET runtime attempts to verify the Authenticode signature by checking against certificate revocation lists (CRLs) at `crl.microsoft.com`. If this site cannot be reached, the .NET framework check will time out (up to 60 seconds). This timeout occurs every time a signed assembly is loaded which can lead to very long load times. You can fix this problem by allowing access to `crl.microsoft.com`. See Microsoft KB article *Microsoft KB article 936707* for background information.

If the computer is not connected to the internet, you can disable CRL checks for the entire system in Internet Explorer. Go to **Options**, select the **Advanced** tab, and under **Settings** clear the **Check for publisher's certification revocation** option.

It is also possible to specify a `generatePublisherEvidence` element in an `<app>.exe.config` that will disable CRL checks for the specific application that you are running. Keep in mind that if you are using Authentication Services components in PowerShell or MMC, you will need to add this configuration for the `powershell.exe.config` and/or `mmc.exe.config`. Refer to *<generatePublisherEvidence> Element* for details.

Pointer Record updates are rejected

If Pointer Record (PTR) updates are being rejected, it may be because the DHCP server is doing the update already. Refer to the documentation for the DHCP server used in your environment. The Microsoft DHCP server does updates on behalf of the host and this is controlled by the FQDN option. Please refer to the Microsoft Active Directory DNS/DHCP documentation.

Resolving DNS problems

It is imperative that DNS is correctly configured. Authentication Services relies on DNS in order to locate domain controllers. Follow these steps to verify that domain controllers can be located using DNS:

1. Use `dig` to test whether your DNS configuration can locate a domain controller. Enter the following at the Unix command prompt, replacing `<DNS Domain Name>` with your Active Directory DNS domain name:

   ```
   dig -t any _ldap._tcp.dc._msdcs.<DNS Domain Name>
   ```

   If DNS is configured correctly, you will see a list of domain controllers for your domain. If not, work with your DNS administrator to resolve the issue.

2. Use `dig` to test whether you can locate a domain controller in your site. Enter the following at the Unix command prompt, replacing `<Site Name>` with the name of your Active Directory site and `<DNS Domain Name>` with your Active Directory DNS domain name.

   ```
   dig -t _ldap._tcp.<Site Name>._sites.dc._msdcs.<DNS Domain Name>
   ```

   If DNS is configured correctly, you will see a list of domain controllers for your site. If not, work with your DNS administrator to resolve the issue.

It is possible to work around DNS problems using the `vastool join` command to specify the domain controller host name on the command line. Authentication Services can work without DNS configured as long as the forward lookup in the `/etc/hosts` file exists. The forward lookup resolves the domain controller host name to an IP address.

You can test this on Linux by firewalling DNS (port 53) with `iptables`. Make sure that you have an entry for your domain controller in `/etc/hosts`, then as root, enter the following commands replacing `<administrator>` with the name of an Active Directory administrator `<DNS Domain Name>` with your Active Directory DNS domain name and `<DC Host Name>` with the host name of your domain controller:

```
iptables -A INPUT -p udp --dport 53 -j DROP
iptables -A OUTPUT -p udp --dport 53 -j DROP
/opt/quest/bin/vastool -u <administrator> join <DNS Domain Name> <DC Host Name>
```

Resolving preflight failures

If one of the `preflight` checks fail, preflight prints a suggested resolution. The following table provides additional problem resolution information. The checks are listed by the associated command-line flags.

Table 28: Install checks

Preflight option	Check	Resolution
--os-patch	Checks for supported operating system and correct operating system patches.	Install the Authentication Services agent on a supported operating system that has the required operating system patches. Click www.oneidentity.com/products/authentication-services/ to view a list of supported Unix and Linux platforms that run Authentication Services.
--disk-space	Checks for sufficient disk space to install Authentication Services.	Free up more disk space. Authentication Services requires disk space in `/opt`, `/etc`, and `/var` to install.

Table 29: Join checks

Preflight option	Check	Resolution
--tld	Checks that the DNS Top Level Domain (TLD) is not '.local'.	Ensure that mDNS is disabled in `/etc/nsswitch.conf` or use a domain other than .local.
--hostname	Checks that the hostname of the system is not 'localhost'.	One Identity recommends that you have a unique hostname in order to maintain uniqueness of computer names in Active Directory. Another option is to ignore this check and use `-n computer_name` when joining. See the *vastool man page* for more information.
--name-service	Checks if the name service is configured to use DNS.	Ensure your host is configured to use DNS properly. Consult your platform documentation to determine the proper method to enable DNS for hostname resolution. See Resolving DNS problems on page 107 for solutions.
--host-resolve	Ensures that the host can	Check your `/etc/resolv.conf` file to ensure that name

Preflight option	Check	Resolution
	resolve names using DNS.	server entries are correct and reachable. Make sure that UDP port 53 (DNS) is open. This check attempts to resolve the domain name and can fail if your DNS configuration is invalid. This check expects to find properly formatted IPv4 addresses. Invalid or unreachable name server entries will cause delays even though the check will pass if at least one valid name server is found. If you notice delays when running this check, make sure that your name server configuration does not reference invalid name servers. See Resolving DNS problems on page 107 for solutions.
--srv-records	Checks for a nameserver that has the appropriate DNS SRV records for Active Directory.	SRV records advertise various Active Directory services. Your configured name server must provide SRV records in order for Authentication Services to take advantage of automatic detection and fail over. Ensure that UDP port 53 (DNS) is open.
--dc	Detects a writable domain controller with UDP port 389 open.	If a domain controller is passed on the preflight command line, `preflight` checks that UDP port 389 is open and that the domain controller is writable. In this case, you may be able to specify a different domain controller. If you do not pass in the name of a domain controller, this check attempts to locate a writable domain controller using DNS SRV records. Ensure that your DNS SRV records are up to date in the configured DNS server. Authentication Services can work with read-only domain controllers, but the computer object must have already been created with the proper settings in Active Directory.
--site	Detects Active Directory site, if available.	This check warns you if Authentication Services was unable to locate an Active Directory site based on your computer's network address. A site configuration is not necessary, but Authentication Services performs better if site information is configured in Active Directory. To resolve this problem, configure a site in Active Directory.
--kerberos-password	Checks if TCP port 464 is open for Kerberos kpasswd.	Ensure that TCP port 464 (kpasswd) is open. This port must be open in order for Authentication Services to set the computer object's password.

Preflight option	Check	Resolution
--kerberos-traffic	Checks if UDP port 88 and TCP port 88 are open for Kerberos traffic.	These ports are the main Kerberos communication channels; they must be open for Authentication Services to authenticate to Active Directory. By default Authentication Services uses TCP, but may be configured to prefer UDP.
--ldap	Checks if TCP port 389 is open for LDAP.	This port must be open for Authentication Services to communicate with domain controllers using LDAP. This communication is GSS SASL encrypted and signed.
--global-catalog	Checks whether the Global Catalog is accessible on TCP port 3268.	Authentication Services can function in a limited way without a global catalog server; however, Authentication Services will be unable to resolve Active Directory users and groups from domains in the forest other than the one to which the host is joined. In addition, some searches may be slower. Make sure that TCP port 3268 (global catalog) is open and that you have configured at least one domain controller as a global catalog and that the global catalog server is up and reachable.
--timesync	Checks the machine's time is not skewed too far from Active Directory.	If the time difference between the Unix host and the domain controller is too large, Kerberos traffic will not succeed. You can usually resolve this failure by running `vastool timesync` to synchronize time with the Active Directory domain. Port 123 UDP must be open in order to synchronize time with the domain controller. This check automatically synchronizes the time if you specify the `-S` option and run the application with root permissions.
--app-configuration	Checks for the Authentication Services application configuration in Active Directory.	This checks fails if you have not configured the Active Directory forest for Authentication Services. Use Control Center (Windows) to create the necessary application configuration. This check can also fail due to an invalid username/password or if there is a time synchronization problem between the Unix host and the domain controller.
--rodc	Checks against the given domain controller even if it is read-only, instead of selecting	The `--rodc` option runs preflight against the given domain controller instead of picking a writable DC. The `--rodc` check affects the `--kerberos-*` and `--ldap` checks. If the `--rodc` check fails, resolve preflight port check failures.

Preflight option	Check	Resolution
	another domain controller.	

NOTE: If you get a message that says `Unable to locate Authentication Services Application Configuration`, you can ignore that error and proceed with the Authentication Services installation. The Authentication Services Active Directory Configuration Wizard starts automatically to help you configure Active Directory for Authentication Services the first time you start the Control Center.

Table 30: Post-join checks

Preflight option	Check	Resolution
--ms-cifs	Checks if TCP port 445 is open for Microsoft Directory Services CIFS traffic.	In order to use Group Policy on Unix, this port must be open to allow Authentication Services to use the CIFS protocol to download Group Policy objects from domain controllers.

Time synchronization problems

Kerberos is a time-sensitive protocol. Your Unix hosts must be synchronized within five minutes of your Active Directory domain controllers. Run the following command as root to have Authentication Services synchronize the local time with Active Directory:

```
vastool timesync
```

System optimization

Kerberos works best with a random-number generator package installed on the operating system. If one is not installed, it will use a potential slow fallback entropy generating system.

HP-UX

HP provides a `/dev/random` driver for hp-UX 11i (11.11), named KRNG11I. It is available, for free, from the KRNG11I depot. You can check if this is already installed by running:

```
$ swlist KRNG11I
```

For older versions (hp-UX 11.00), an open-source implementation of `/dev/random` is available from *"random" DLKM (dynamically loadable kernel module) for HP-UX* .

Solaris

Entropy is generally obtained from `/dev/random`, which is an interface to a kernel random source. On Solaris 8, the `/dev/random` driver is provided in the following patches from *ORACLE*:

- solaris8/sparc: OS patch 112438
- solaris8/x86: OS patch 112439

Unable to install or upgrade

The most common installation or upgrade failure is that the Unix host cannot read the Authentication Services application configuration in Active Directory. Ensure that you have followed the instructions in Configure Active Directory for Authentication Services on page 44 and that the configuration has been created successfully.

During an upgrade, you may see an error that Authentication Services cannot upgrade because the application configuration cannot be located. If you previously joined to a specific domain controller, Authentication Services disabled DNS SRV record lookups. This means that Authentication Services cannot resolve other domains in the forest and may be unable to locate the application configuration. In this case, you must ensure that the domain controller you specified is a global catalog. Otherwise, you must create the Authentication Services application configuration in the domain that you join or you must properly configure DNS to return SRV records and join normally, rather than specifying a domain controller when you join.

For more information, see About Active Directory configuration on page 46.

Unable to join the domain

If you are unable to join the domain, run the `preflight` utility to validate your environment.

For more information, see *The Authentication Services Pre-Installation Diagnostic Tool* in the *Authentication Services Installation Guide*.

Then, verify the following:

- Check that the Active Directory account specified during join has rights to join the computer to the domain.
- Check that the Unix host is able to properly resolve the domain name through DNS.

If you are joining to a specific domain controller you must ensure that Authentication Services can locate and read the configuration information in Active Directory. You should do one of the following:

- Make sure the domain controller you specify is a global catalog.
- Create the Authentication Services application configuration in the domain to which you are joining.

  For more information, see About Active Directory configuration on page 46.
- Properly configure DNS to return srv-records and avoid joining to a specific domain controller.

Unable to log in

If you are unable to log in as an Active Directory user after installing, check the following:

1. Log in as root on the Unix host.
2. Check the status of the Authentication Services subsystems. To do this, run the following command:

   ```
   vastool status
   ```

   Correct any errors reported by the status command, then try logging in again.
3. Ensure the user exists locally and is allowed to log in. To check this, run the following command:

   ```
   vastool user checklogin <username>
   ```

   The output displays whether the user is a known Active Directory user. If not, you may need to map the user to an Active Directory account or Unix-enable the Active Directory account. If the user is known, an access control rule may prevent them from logging in. The output of the command displays which access control rules are in effect for the user.

You may need to restart window managers such as `gdm` in order for the window manager to reload NSS modules. Until the window manager reloads the NSS configuration, you will be unable to log in with an Active Directory user. Other services such as `cron` may also be affected by NSS changes. If you are unsure which services need to be reloaded, reboot the system.

NOTE:

If you are configuring on VMware ESX Server vSphere (ESX 4.0) the reason you can not log in may be related to access control issues. See *Configuring Access Control on ESX 4* in the *Authentication Services Administration Guide*.

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

O

P

Q

R

S

T

U

V

W