# One Identity Authentication Services 4.1.7

## Evaluation Guide

Authentication Services Evaluation Guide
Updated - November 2018
Version - 4.1.7

# 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 a breeze. 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 is a one-stop shop for Unix security that 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: **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 enables 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 host(s) 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 either:

   - **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

Welcome to the *Authentication Services Evaluation Guide*.

This is a self-directed, hands-on evaluation of Authentication Services. The content includes a product overview, installation instructions, and a "Getting Started" section that will help you get acquainted with the Control Center, and how to use Authentication Services to accomplish basic system administration tasks.

The guide is divided into three sections:

- Introducing One Identity Authentication Services on page 9
- Installing and configuring Authentication Services on page 21
- Getting started with Authentication Services on page 42

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

# 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.

## Licensing Authentication Services

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

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:<br>• Windows XP SP2 (or later)<br>• Windows Vista<br>• Windows 7<br>• Windows 8<br>• Windows Server 2003 SP1 (or later)<br>• Windows Server 2008<br>• Windows Server 2008 R2<br>• Windows Server 2012<br>• Windows Server 2012 R2<br>• Windows Server 2016<br>• Windows Server 2019<br>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:<br>• Windows Installer 3.1 (http://support.microsoft.com/kb/893803)<br>• Microsoft .NET Framework 3.5 SP1 or higher<br>• Windows PowerShell 1.0 or higher (http://support.microsoft.com/kb/968929)<br>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-Shell cmdlets | Provides the ability to script Unix management tasks. |
| Documentation | Full product documentation and online help. |

# 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*, *showInAdvancedViewOnly* in the application configuration location.

**Table 3: Required Windows permissions**

| Rights Required | For User | Object Class | Attributes |
|---|---|---|---|
| Create | Authentication Services | Container | |

| Rights Required | For User | Object Class | Attributes |
|---|---|---|---|
| Child Object | Administrators Only | | |
| 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: Patch level requirements**

| Platform | Patch Level |
|---|---|
| Solaris 8 SPARC | 108993-55 or greater |
| Solaris 8 X86 | 108994-01 or greater<br>112757-01 or greater |
| Solaris 9 SPARC | 112874-37 or greater<br>112960-14 or greater<br>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 |

| Platform | Patch Level |
|---|---|
| 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<br>GOLDAPPS11i quality packs<br>BUNDLE11i - Patch bundle<br>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 | 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. |

| Unix Component | Description |
|---|---|
| Tool) | |
| 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 | NA |
| Authentication Services Application Configuration: changes<br>• Unix Global Settings<br>• Licensing<br>• Custom Unix Attributes | Update permission to the containers created above (no particular permissions if you are the one who created it) | NA |
| Schema optimization | Schema Administrator rights | NA |
| Display Specifier Registration | Enterprise Administrator rights | NA |

| Function | Active Directory Permissions | Local Client Permissions |
|---|---|---|
| Editing Users | Administrator rights | NA |
| Create any group policy objects | Group Policy Creator Owners rights | NA |
| RFC 2307 NIS Import Map Wizard | Location in Active Directory with Create Container Object rights (you create containers for each NIS map) | NA |
| Unix Account Import Wizard | Administrator rights (you are creating new accounts) | NA |
| Logging Options | Write permissions to the file system folder where you want to create the logs | NA |
| `vasd` daemon | The client computer object is expected to have read access to user and group attributes, which is the default.<br>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 operatingSystemServicePack**. | `vasd` must run as root |
| QAS/VAS PAM module | NA (updated by means of `vasd`) | Any local user |
| QAS/VAS NSS module<br>`vastool nss` | NA (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`<br>`vastool unjoin` | computer creation or deletion permissions in the desired container | root |
| `vastool configure`<br>`vastool unconfigure` | NA | root |
| `vastool search`<br>`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 |

| Function | Active Directory Permissions | Local Client Permissions |
|---|---|---|
| | | user |
| `vastool cache` | NA | 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 |
| `vastool group add`<br>`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 }` | NA | 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`<br>`vastool inspect` | NA | Any local user |

| Function | Active Directory Permissions | Local Client Permissions |
|---|---|---|
| `vastool license` | | |
| `vastool kinit`<br>`vastool klist`<br>`vastool kdestroy` | local client needs permissions to modify the keytab specified, default is the computer object which is root. | Any local user |
| `vastool ktutil` | NA | 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`<br>`vastool unmerge` | NA | root |
| `vastool passwd` | Regular Active Directory user | Any local user |
| `vastool passwd <AD user>` | Active Directory user with password reset permission | Any local user |
| `vastool schema list`<br>`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 | NA |
| `vastool smartcard` | NA | root |
| `vastool status` | NA | root |

| Function | Active Directory Permissions | Local Client Permissions |
|---|---|---|
| `vastool timesync` | NA | 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 }` | NA | 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 which 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:<br>• Windows x86 (32-bit)<br>• Windows x86-64 (64-bit)<br>• Unix/Linux systems for which Java 8 is available |
| Server Requirements | 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 Requirements | Click www.oneidentity.com/products/authentication-services/ to view a list of Unix, Linux, and Mac platforms that support Authentication Services.<br>Click www.oneidentity.com/products/privilege-manager-for-unix/ to review a list of Unix and Linux platforms that support Privilege Manager for Unix.<br>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.<br>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.<br>NOTE: Management Console for Unix does not support Security-Enhanced Linux (SELinux)<br>NOTE: When you install Authentication Services on Solaris 10 (SPARC - 32/64-bit), the Solaris 10 packages are installed. |
| Default Memory Requirement: | 1024 MB |

| Component | Requirements |
|---|---|
| | 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 the Active Directory site membership. |
| 3268 | Used for LDAP searches against Active Directory Global Catalogs. TCP is always used when searching against the Global Catalog. |
| 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.

# Installing and configuring Authentication Services

To extend the authentication, authorization, and administration infrastructure of Active Directory to the rest of your enterprise, allowing Unix, Linux, and Mac OS X systems to act as full citizens within Active Directory, you must install and configure Authentication Services.

This section explains the steps you must take in detail:

1. Install Management Console for Unix.
2. Install Authentication Services Windows components.
3. Configure Active Directory for Authentication Services (one time, only).
4. Configure Unix Agent Components
    a. Configure the mangement console for Active Directory.
    b. Prepare the Unix hosts for Active Directory user access:
        - Add and profile a host.
        - Check the host for readiness to join Active Directory.
        - Install Authentication Services agent software packages on the host to allow Active Directory user access.

            NOTE: For users to authenticate on Unix, Linux, and Mac OS X hosts with Active Directory credentials, your Unix hosts must have the Authentication Services agent installed.

        - Join the host to Active Directory.

## Install the management console

In preparing for your Authentication Services installation, One Identity recommends that you install Management Console for Unix. 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.

You can install the mangement console on Windows, Unix, or Linux computers. Each hosting platform prompts for similar information.

The following install files are located on the Authentication Services distribution media under `console | server`:

- `ManagementConsoleForUnix_unix_2_n_n.sh` - for Unix and Linux
- `ManagementConsoleForUnix_windows_2_n_n.exe` - for Windows
- `ManagementConsoleForUnix_windows-x64_2_n_n.exe` - for Windows

where "n.n" indicates the product version number.

The *Management Console for Unix Administrator's Guide* contains detailed instructions for installing the mangement console on all of these platforms. Use the following procedure to install the console on a supported Windows platform from the Authentication Services 4.1 distribution media.

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 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. On 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.

# Install Authentication Services Windows components

One Identity recommends that you install the Windows components and configure Active Directory before you install the Unix components.

# Installing Authentication Services Windows components

Install Authentication Services on each Windows Workstation you plan to use to administer Unix data in Active Directory.

***To install the Authentication Services Windows components***

1. From the Autorun Setup tab, click **Authentication Services** to launch the setup wizard.
2. At the Software License Agreement dialog, accept the terms of the End User License Agreement and click **Install**.

The Authentication Services Setup wizard installs all Authentication Services components by default.

To only install specific components, click the **Customize installation options** link. (For more information, see *Customize Installation Options* in the *Authentication Services Installation Guide*.)

3. Once the installation completes successfully, click **Finish** or **Launch Control Center**.

# 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* in the *Authentication Services Installation Guide*.)

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* in the *Authentication ServicesInstallation Guide*.)

***To configure Active Directory for Authentication Services***

1. At the Authentication Services Active Directory Configuration Wizard Welcome dialog, click **Next**.
2. At 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. At the License Authentication Services dialog, browse to select your license file and click **Next**.

   Refer to Licensing Authentication Services on page 9 for more information about licensing requirements.

   NOTE: You can add additional licenses later from the Authentication Services Control Center Preferences Licensing dialog.

4. At 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 11.

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.

   (Refer to *Configure Unix Agent Components* in the *Authentication Services Installation Guide* for more information.)

# 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.

(See *Version 3 Compatibility Mode* in the *Authentication Services Installation Guide* for details.)

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 are 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.

This table summarizes the required rights.

**Table 10: 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.

# 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

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 on page 30.

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 the console for Active Directory logon

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. On 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

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. On 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. On 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 on 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 set up, 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

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. On 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 consoles 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. On 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. On the **Summary** dialog, click **Finish**.

   The Management Console for Unix login screen opens.

# Management Console for Unix log on page

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 online Help for details.) To add additional accounts to this access list, see *Add (or Remove) Role Members* in 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.

# Add 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. On 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 host (s) 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 on 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 Profile hosts on page 34 which walks you through the host profile steps.
6. If you clear the **Profile hosts after adding** option on 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 on the **Add Hosts** dialog.

# Profile 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 on 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 host(s).

   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 host(s).
   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 log on 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 log on 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 (that 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 checked. This enables the mangement console to automatically accept 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 key(s) 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 new/changed key(s). 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/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.

## Profile automatically

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 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 on 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-profile' turned on, or they all have 'Auto-profile' 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, either:
   - **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** on 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 login 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 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 recreate the "questusr" account***

a. Re-profile the host.

b. Reconfigure the host for automatic profiling.

5. On the Log on to Host dialog, enter the user credentials to access the selected host(s) 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 host(s) and click **OK**.

b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, it displays a grid which 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. On the Log on to Host dialog, enter the user credentials to access the selected host(s) 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.

# Check 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 host(s) 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. On the Log on to Host dialog, enter the user credentials to access the selected host(s) 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

host(s) and click **OK**.

b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, it displays a grid which 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.

# Install 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. On 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 which 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 which 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 which 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 which 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. On the Log on to Host dialog, enter the user credentials to access the selected host(s) 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 host(s) and click **OK**.

   b. If you selected multiple hosts and the Enter different credentials for each selected host option, it displays a grid which 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.

# Join hosts to Active Directory

In order to manage access to a host using Authentication Services for Active Directory, you must join the host to an Active Directory domain. Joining a host to a domain creates a computer account for that host. Once you have deployed and installed the Authentication Services Agent software on a host, use the **Join to Active Directory** command on the All Hosts view's **Join** menu to join the host to an Active Directory domain.

***To join hosts to Active Directory***

1. Select one or more hosts from the list on the All Hosts view, open the **Join or Configure** menu tool bar button and select **Join to Active Directory**.

   NOTE: The **Join to Active Directory** tool bar menu is enabled when you select hosts that have the Authentication Services Agent installed and are not joined Active Directory.

   The tool bar button will not be active if:

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

2. On the Join Host to Active Directory dialog, enter the following information to define how and where you want to join the host to Active Directory:

   a. Select the Active Directory domain to use for the join operation or enter the FQDN of the Active Directory domain.

      Use the same domain you entered when you performed the Check for AD Readiness.

   b. Optionally enter a name for the computer account for the host.

      Leave this field blank to generate a name based on the host's DNS name.

   c. Click the button to locate and select a container in which to create the host computer account.

   d. Enter the optional join commands to use.

      See *Optional Join Commands* in the mangement console online Help for a list of commands available.

   e. Enter the user name and password to log onto Active Directory.

      The user account you enter must have elevated privileges in Active Directory with rights to create a computer account for the host.

3. On the Log on to Host dialog, enter the user credentials to access the selected host(s) and click **OK**.

   NOTE: This task requires elevated credentials. The mangement console pre-populates this information.

   The Task Progress pane on the All Hosts view displays a progress bar and the final status of the tasks, including any failures or advisories encountered.

# 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 11: 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 GPO's 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 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 enables 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.

# Filter options

***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.

# Edit 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 ServicesAdministration Guide*, located in Control Center **Tools** view in the *Documentation* section, or in the `docs` directory of the installation media.

# 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 Unix settings report***

1. From the Group Policy 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.

# Show 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.

# Launch 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

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 9 for more information about licensing requirements.)

### Add 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**.

# Display specifiers

Display specifiers are Active Directory objects which provide information about how other objects in the directory display in client applications.

## Register display specifiers

Because it is common to use the Find dialog in ADUC to manage users and groups, One Identity recommends that you register display specifiers with Active Directory. Registering display specifiers provides the following benefits:

- Unix Account properties appear in ADUC Find dialog results.
- Unix Personality objects are displayed correctly in ADUC. This only applies if the Unix Personality schema has been installed.

NOTE: You must have Enterprise Administrator rights to register display specifiers.

You can inspect exactly what changes are made during the display specifier registration process by viewing the `DsReg.vbs` script found in the Authentication Services installation directory. You can use this script to unregister display specifiers at a later time.

***To register display specifiers with Active Directory***

1. From a Windows management workstation with Authentication Services installed, navigate to **Start | Quest Software | Authentication Services | Control Center**.
2. Click **Preferences** on the left navigation panel.
3. Expand the **Display Specifiers** section.

   NOTE: The **Register Display Specifiers** link displays only when display specifiers are not already registered with Active Directory. If the display specifiers are registered, Control Center does not display the link.

4. Click the **Register Display Specifiers** link to register display specifiers with Active Directory.

   While it is registering the display specifiers with Active Directory, Control Center displays a progress indicator. When the process is complete Control Center indicates that display specifiers are registered.

   Alternatively, you can register display specifiers from the command line, as follows:

   a. Log in as a user with Enterprise Administrator rights.

   b. Open a command prompt, navigate to the Authentication Services installation directory, and run this command:

   ```
   DsReg.vbs /add
   ```

NOTE: To register One Identity Active Roles Server display specifiers with One Identity Active Roles Server, navigate to the installed location for Authentication Services and run the following command:

```
DsReg.vbs /add /provider:EDMS
```

You must install the One Identity Active Roles Server management package locally or `DsReg.vbs` returns an "Invalid Syntax" error.

To see all the `DsReg.vbs` options, run the following command:

```
DsReg.vbs /help
```

## Unregistering display specifiers

NOTE: You must have Enterprise Administrator rights to unregister display specifiers.

***To unregister display specifiers in Active Directory***

1. Log in as a user with Enterprise Administrator rights.
2. Open a command prompt and navigate to the Authentication Services installation directory.
3. Run the `DsReg.vbs` script with the `/remove` option:

   ```
   DsReg.vbs /remove
   ```

   NOTE: To unregister display specifiers with One Identity Active Role, run the following command:

   ```
   DsReg.vbs /remove /provider:EDMS
   ```

   To see all the `DsReg.vbs` options, run the following command:

   ```
   DsReg.vbs /help
   ```

   A SUCCESS message appears indicating that the display specifiers were removed successfully.

# 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 log in.

**Table 12: 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 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 13: 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 | Enter the minimum value for the Unix Group ID (GID). Typically |

| Option | Description |
|---|---|
| Number | 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 14: 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 which 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. |
| 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.

## Enable 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 15: 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, please note, 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 16: 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.

## Configure 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 53.

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 32.)

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.

# Add 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 *Host* tab's *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.

# Add 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 Add a local group on page 54.

    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. On the Log on to Host dialog, enter your credentials to log onto 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.

# Add 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**.

# Add 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.

# Change 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.

## Enable 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 into 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 *Host* tab's *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 Add local user account on page 55.
3. On the *AD Logon* tab, select the **Require an AD Password to logon to Host** option, and click **Select**.
4. On 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 Add an Active Directory user account on page 56.
5. On the localuser's properties, click **OK**.

6. On the Log on to Host dialog, verify your credentials to log onto 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. On the Log on to Host dialog, verify your credentials to log onto the host and click **OK**.

   NOTE: This task requires elevated credentials.

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

# Test the mapped user login

Once you have "mapped" a local user to an Active Directory user, you can log into 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:
   - the Unix host name in the **Host name** box
   - the local user name, **localuser**, in the **User name** box

   and click **Login** to log onto 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-enable 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 Add an Active Directory group account on page 55.
4. On the *Unix Account* tab, select the **Unix-enabled** option and click **OK**.

# Unix-enable 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* log on 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 73 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.

# Test the Active Directory user login

Now that you have Unix-enabled an Active Directory user, you can log into 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:
   - the Unix host name in the **Host name** box
   - the Active Directory user name, such as **ADuser**, in the **User name** box

   and click **Login** to log onto 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.

# Run 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 Profile automatically on page 35.)

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 which 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 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 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 online Help for details.) For example, you must have an activated policy server to activate the sudo-related reports.

## Host reports

**Table 17: 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:<br>• Total number of hosts<br>• Total number, percentage and names of the hosts ready to join<br>• Total number, percentage and names of the hosts ready to join with advisories |

| Report | Description |
| --- | --- |
| | • Total number, percentage and names of the hosts not ready to join<br>• Total number of hosts not checked for AD readiness<br><br>Use the following report parameters to define details to include in the report.<br><br>• Joined to AD<br>• Ready to Join AD<br>• Ready to Join AD with Warnings<br>• Not Ready to Join AD<br>• Not Checked for Readiness<br><br>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:<br><br>• Total number of hosts<br>• Total number, percentage and names of the hosts ready to join<br>• Total number, percentage and names of the hosts not ready to join<br>• Total number of hosts not checked for readiness<br><br>Use the following report parameters to define details to include in the report.<br><br>• Joined to a policy group<br>• Ready to join a policy group<br>• Ready to join a policy group with warnings<br>• Not ready to join a policy group<br>• Not checked for readiness<br><br>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.<br><br>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. |

| Report | Description |
|---|---|
| | 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:<br>• Total number of hosts included in the report<br>• Host Name, IP Address, OS, Hardware<br>• Sudo version number<br>Use the following report parameters to define details to include for each host.<br>• Authentication Services Properties<br>• Privilege Manager Properties<br>• Local Users<br>• Local Groups<br>• Host SSH Keys<br>• Installed One Identity Software<br>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

**Table 18: User reports**

| Report | Description |
|---|---|
| AD User Conflicts | Returns all users with Unix User ID numbers (UID numbers) assigned to other Unix-enabled user accounts.<br>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.<br>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:<br>• Host Name, DNS Name or IP Address where a conflict would occur |

| Report | Description |
| --- | --- |
| | • User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory and Login Shell for each host where conflicts exist<br><br>Use the following report parameters to define the user name and UID number that would cause a conflict with existing local user accounts:<br><br>• User Name is<br>• UID Number is<br><br>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:<br><br>• Host Name, DNS Name or IP Address where the user exists<br>• User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory, and Login Shell for each host where the user exists<br><br>If you do not define a specific user, it includes all local users on each profiled host in the report.<br><br>To locate a specific user, use the following report parameters:<br><br>• User Name contains<br>• UID Number is<br>• Primary GID Number is<br>• Comment (GECOS) contains<br>• Home Directory contains<br>• Login Shell contains<br><br>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.<br><br>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:<br><br>• Host Name, DNS Name or IP Address of hosts where users exist that are required to log on using their AD credentials<br>• User Name, UID Number, Primary GID Number and Comment (GECOS) of local user account |

| Report | Description |
| --- | --- |
| | • The SAM account Name of the Active Directory account that the local user account must use to log on<br><br>NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent.<br><br>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:<br>• Username<br>• Empty password<br>• UID<br>• GID<br>• GECOS<br>• Home directory path<br>• Account's shell<br>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 users to Active Directory.<br>Indicate how you want to match user accounts by selecting the value parameters that you want to match:<br>• Username<br>• UID<br>• GID<br>• GECOS<br>• Home Directory<br>• Shell<br>Optionally, you can include the host name for the accounts, as well:<br>• Include the host name for accounts |

| Report | Description |
| --- | --- |
| | 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.<br><br>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.<br><br>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.<br><br>NOTE:<br><br>• 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.<br>• If Login Shell is `/bin/false`, the user is considered to be disabled for Unix or Linux logon.<br>• Account Disabled indicates whether the Active Directory User account is enabled or disabled.<br><br>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.<br><br>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 online help), and you are logged on as an Active Directory account in the *Manage Hosts* role. |

## Group reports

**Table 19: Group reports**

| Report | Description |
|---|---|
| AD Group Conflicts | Lists all Active Directory groups with Unix Group ID (GID) numbers assigned to other Unix-enabled groups.<br>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.<br>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:<br>• Host Name, DNS Name or IP Address where the group exists<br>• Group Name, GID Number, and members for each host where the group exists<br>If you do not specify a group, it includes all local groups on each profiled host in the report.<br>To locate a specific group, use the following report parameters:<br>• **Group Name contains**<br>• **GID Number is**<br>• **Member contains**<br>• **Include all group members in report**<br>NOTE: The **Member contains** field accepts multiple entries separated by a comma. Spaces are taken literally in the search. For example, entering:<br>• **adm, user** searches for members whose name contains 'adm' or ' user'<br>• **adm,user** searches for members whose name contains 'adm' or 'user'.<br>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.<br>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. |

| Report | Description |
|---|---|
| | 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.<br>NOTE: A Group object is considered 'Unix-enabled' if it has a value for the GID Number.<br>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.<br>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 online help), and you are logged on as an Active Directory account in the *Manage Hosts* role. |

## Access & Privileges reports

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

**Table 20: 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:<br>• Total number of users that can log on to the host<br>• The users that can log on to the host<br>• The commands users can run on the host<br>• The runas aliases for which the user can run commands on the host<br>• The commands the runas alias can run on the host<br>**Browse** to select a host.<br>Optionally, select the **Show detailed report** option.<br>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. |

| Report | Description |
| --- | --- |
| Access & Privileges by User | Identifies the users with log-on 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:<br>• Total number of hosts where the user can logon<br>• The hosts where the user can logon<br>• The commands the user can run on each host<br>• The runas aliases for which the user can run commands on each host<br>• The commands the runas alias can run on each host<br>Use the following report parameters to specify the user to include in the report:<br>• A local user (default)<br>• An AD user<br>**Browse** to select a user.<br>Optionally select the **Show detailed report** option.<br>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:<br>• Command name<br>• User who executed the command<br>• Date and time the command was executed<br>• Host where the command was executed<br>Use the following report parameters to define details in the report:<br>• Policy Group<br>• Command<br>• Host<br>• Log status<br>• Date |

| Report | Description |
| --- | --- |
| | NOTE: You can use wildcards in the text string you enter in the **Command** box, such as * and ?.<br><br>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:<br>• List of roles<br>• List of permissions assigned to each role<br>• List and number of members assigned to each role<br><br>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 log-on permission. This report includes the following information for hosts joined to an Active Directory domain:<br>• Total number of hosts where the AD user has access<br>• List of hosts where the AD user has access<br>Specify the Active Directory users to include in the report:<br>• All AD users (default)<br>• Select AD user<br>**Browse** to search Active Directory to locate and select an Active Directory user.<br><br>NOTE: The report might show both the Active Directory login name and local user name(s) 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.<br><br>NOTE: Only hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent are included in this report. |

| Report | Description |
| --- | --- |
| | 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:<br>• Host Name, DNS Name or IP Address of the host selected for the report<br>• Users that have been granted permission to log on<br>Specify the managed hosts to include in the report:<br>• All profiled hosts (default)<br>• Select host<br>**Browse** to locate and select a managed host that is joined to Active Directory.<br>NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.*x* agent.<br>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:<br>• Name of the user that made changes to the policy<br>• Version number for the changes<br>• Time and date the changes were saved and actively used to enforce policy<br>• Changes made to the policy based on version<br>Select a policy group.<br>Select to:<br>• Show all changes to the policy<br>• Show only changes for a specific pmpolicy file (not available for sudo-based policy)<br>• Show changes to the policy for changes for one or more revisions<br>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 reports

**Table 21: 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:<br>• Product<br>• Purchased licenses<br>• Used licenses |

# Use Authentication Services PowerShell

Authentication Services includes PowerShell modules which 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-enable a user and user group

***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 Add an Active Directory group account on page 55.)

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 completely clear all Unix attribute information, enter
>
> `Clear-QasUnixUser ADuser`

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

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

   and click **Login** to log onto 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 22: 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 |

| cmdlet Name | Description |
|---|---|
| | 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 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. |

| cmdlet Name | Description |
|---|---|
| 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 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

## Install 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 the `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.
6. Read the *Change Auditor Installation Guide* to obtain detailed steps for installing Authentication Services Defender.

# 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.

## Install Defender

In order to use strong authentication you must download and install Defender.

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

***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. Read the *Defender Installation Guide* to obtain detailed steps for installing Authentication Services Defender.
8. 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.

# 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



## W