



One Identity Authentication Services
4.2.4

Installation Guide

Copyright 2020 One Identity LLC.

ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of One Identity LLC .

The information in this document is provided in connection with One Identity products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of One Identity LLC products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, ONE IDENTITY ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ONE IDENTITY BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ONE IDENTITY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. One Identity makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. One Identity does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

One Identity LLC.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our Web site (<http://www.OneIdentity.com>) for regional and international office information.

Patents

One Identity is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <http://www.OneIdentity.com/legal/patents.aspx>.

Trademarks

One Identity and the One Identity logo are trademarks and registered trademarks of One Identity LLC. in the U.S.A. and other countries. For a complete list of One Identity trademarks, please visit our website at www.OneIdentity.com/legal. All other trademarks are the property of their respective owners.

Legend

 **WARNING:** A WARNING icon highlights a potential risk of bodily injury or property damage, for which industry-standard safety precautions are advised. This icon is often associated with electrical hazards related to hardware.

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

Contents

Privileged Access Suite for Unix	7
About this guide	8
Introducing One Identity Authentication Services	10
About Authentication Services licenses	10
System requirements	10
Windows and cloud requirements	11
Authentication Services Windows components	12
Windows permissions	12
Unix agent requirements	13
Authentication Services Unix components	14
Authentication Services permissions matrix	15
Authentication Services encryption types	20
Management Console for Unix requirements	20
Network requirements	21
Installing and configuring Authentication Services	23
Install the management console	23
Installing and configuring the management console	24
Install Authentication Services Windows components	25
Installing Authentication Services Windows components	25
Customizing installation options	26
Installing using msiexec.exe	26
Configure Active Directory for Authentication Services	29
Configuring Active Directory for Authentication Services	29
About Active Directory configuration	31
Join the host to AD without the Authentication Services application configuration ..	32
Version 3 Compatibility Mode	33
Configure Unix agent components	33
Set up Management Console for Unix wizard	34
Configure Console for Active Directory Logon dialog	34
Set up console access by role dialog	35
Identify Console dialog	36

Set Supervisor Password dialog	36
Summary dialog	36
Logging in to Management Console for Unix	37
Prepare Unix hosts	37
Adding hosts to the management console	38
Profiling hosts	39
Installing software on hosts	43
Checking readiness	45
Joining hosts to Active Directory	45
Check QAS agent status	46
Installing and joining from the Unix command line	50
The Authentication Services pre-installation diagnostic tool	50
Running preflight	52
The Authentication Services install script	52
Installing the Authentication Services agent	53
Installation script options	53
Licensing Authentication Services	55
Verifying Authentication Services license information	56
Installing licenses from the command line	56
Creating the application configuration from the Unix command line	57
Changing the schema configuration mode	57
Joining the domain	58
Joining the domain using VASTOOL	58
Joining the domain using VASJOIN script	59
Dynamic DNS update tool	60
Getting started with Authentication Services	62
Getting acquainted with the Control Center	62
Management console	63
Group Policy	64
Filtering the list of GPOs	64
Editing a GPO	64
Generating a settings report	64
Showing files	65
Launching GPMC	65

Tools	65
Preferences	66
Licensing	66
Global Unix Options	67
Logging Options	69
Custom Unix Attributes	69
Learning the basics	72
Adding a local group	72
Adding a local user account	73
Adding an Active Directory group account	74
Adding an Active Directory user account	74
Changing the default Unix attributes	75
Active Directory account administration	75
Enabling local user for AD authentication	75
Testing the mapped user login	76
Unix-enabling an Active Directory group	77
Unix-enabling an Active Directory user	77
Testing the Active Directory user login	78
Running reports	78
Reports	80
Use Authentication Services PowerShell	91
Unix-enabling a user and user group (PowerShell Console)	91
PowerShell cmdlets	93
Change Auditor for Authentication Services	95
Installing Change Auditor for Authentication Services	95
One Identity Defender	96
Installing Defender	96
Appendix: Troubleshooting	98
Getting help from technical support	98
Disaster recovery	99
Long startup delays on Windows	99
Pointer Record updates are rejected	99
Resolving DNS problems	99
Resolving preflight failures	100
Time synchronization problems	104

Unable to install or upgrade	104
Unable to join the domain	104
Unable to log in	105
vasypd has unsatisfied dependencies	106
Appendix: Enterprise package deployment	107
Installing the Authentication Services agent package	107
Authentication Services agent upgrade commands	109
Restarting Authentication Services services	111
Uninstalling the Authentication Services agent packages	111
Oracle Solaris 10 zones/containers support	112
Authentication Services and Oracle Solaris 10 Zones installation guidelines	112
About us	114
Contacting us	114
Technical support resources	114
Index	115

Privileged Access Suite for Unix

Unix security simplified

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

Active Directory bridge

Achieve unified access control, authentication, authorization, and identity administration for Unix, Linux, and macOS 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 macOS. See www.oneidentity.com/products/authentication-services/ for more information about the Active Directory Bridge product.

Root delegation

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

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

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

- By choosing to replace sudo, you will still be able to delegate the Unix root privilege based on centralized policy reporting on access rights, but with a more granular permission and the ability to log keystrokes on all activities from the time a user logs

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

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

Privileged Access Suite for Unix

Privileged Access Suite for Unix offers two editions: *Standard* edition and *Advanced* edition. Both editions include the Management Console for Unix, a common management console that provides a consolidated view and centralized point of management for local Unix users and groups; and Authentication Services, patented technology that allows organizations to extend the security and compliance of Active Directory to Unix, Linux, and macOS 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 management console.
3. Add and profile hosts using the management console.
4. Configure the console to use Active Directory.
5. Deploy client software to remote hosts.

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

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

About this guide

The *Authentication Services Installation Guide* is intended for Windows, Unix, Linux, and Macintosh system administrators, network administrators, consultants, analysts, and any other IT professionals who will be installing and configuring Authentication Services for the first time. This guide walks you through the process of installing, upgrading, and uninstalling the Authentication Services agent.

| 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 macOS 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.

About Authentication Services licenses

Authentication Services must be licensed in order for Active Directory users to authenticate on Unix and macOS 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 a valid Authentication Services license installed for full functionality.

NOTE: In order to use Starling Two-Factor Authentication with Authentication Services, you must have a valid license for Authentication Services with One Identity Hybrid Subscription included.

To obtain a license, use the [Licensing Assistance](#) page on the One Identity support page or contact your account representative.

For more information on installing Authentication Services licenses, see [Licensing Authentication Services](#).

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 and cloud requirements

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

Table 1: Authentication Services requirements

System requirements	
Supported Windows Platforms	<p>Prerequisite Windows software</p> <p>If the following prerequisite is missing, the Authentication Services installer suspends the installation process to allow you to download the required component. It then continues the install:</p> <ul style="list-style-type: none">• Microsoft .NET Framework 4.5 <p>You can install Authentication Services on 64-bit editions of the following configurations:</p> <ul style="list-style-type: none">• Windows Server 2008 R2• Windows Server 2012• Windows Server 2012 R2• Windows Server 2016• Windows Server 2019 <p>NOTE: Due to tightened security, when running Authentication Services Control Center on Windows 2008 R2 (or later) 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.</p>
Supported cloud services	<ul style="list-style-type: none">• AWS Directory Service for Microsoft Active Directory (also called AWS Managed Microsoft AD)• Azure Active Directory Domain Services• Google Cloud Platform Managed Service for Microsoft Active Directory

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 macOS.
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`, and `showInAdvancedViewOnly` in the application configuration location.

Table 3: Required Windows permissions

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

Unix agent requirements

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

NOTE: With Authentication Services 4.2 and later, Linux platforms require glibc 2.4 (or later).

The following table provides a list of supported Unix and Linux platforms for Authentication Services.

Table 4: Unix agent: Supported platforms

Platform	Version	Architecture
Amazon Linux AMI		x86_64
Apple macOS	10.12, 10.13, 10.14, 10.15	x86_64
CentOS Linux	5, 6, 7, 8	Current Linux architectures: s390, s390x, PPC64, PPC64le, ia64, x86, x86_64, AARCH64
Debian	Current supported releases	x86_64, x86, AARCH64
Fedora Linux	Current supported	x86_64, x86, AARCH64

Platform	Version	Architecture
	releases	
FreeBSD	10.x, 11.x, 12.x	x32, x64
HP-UX	11.31	PA, IA-64
IBM AIX	7.1, 7.2	Power 4+
OpenSuSE	Current supported releases	x86_64, x86, AARCH64
Oracle Enterprise Linux (OEL)	5, 6, 7, 8	Current Linux architectures: s390, s390x, PPC64, PPC64le, ia64, x86, x86_64, AARCH64
Oracle Solaris	10 8/11 (Update 10), 11.x	SPARC, x64
Red Hat Enterprise Linux (RHEL)	5, 6, 7, 8	Current Linux architectures: s390, s390x, PPC64, PPC64le, ia64, x86, x86_64, AARCH64
SuSE Linux Enterprise Server (SLES)/Workstation	11, 12, 15	Current Linux architectures: s390, s390x, PPC64, PPC64le, ia64, x86, x86_64, AARCH64
Ubuntu	Current supported releases	x86_64, x86, AARCH64

NOTE: For maximum security and performance, before you begin the installation, make sure that you have the latest patches for your operating system version. One Identity recommends that you run the Preflight utility to check for supported operating systems and correct operating system patches.

For more information, see [Running preflight](#) on page 52.

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

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

Authentication Services permissions matrix

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

Table 6: Authentication Services: Required permissions

Function	Active Directory permissions	Local client permissions
Authentication Services Application Configuration: creation	Location in Active Directory with Create Container Object rights	N/A
Authentication	Update permission to the containers created above	N/A

Function	Active Directory permissions	Local client permissions
Services Application Configuration: changes <ul style="list-style-type: none"> • Unix Global Settings • Licensing • Custom Unix Attributes 	(no particular permissions if you are the one who created it)	
Schema optimization	Schema Administrator rights	N/A
Display Specifier Registration	Enterprise Administrator rights	N/A
Editing Users	Administrator rights	N/A
Create any group policy objects	Group Policy Creator Owners rights	N/A
RFC 2307 NIS Import Map Wizard	Location in Active Directory with Create Container Object rights (you create containers for each NIS map)	N/A
Unix Account Import Wizard	Administrator rights (you are creating new accounts)	N/A
Logging Options	Write permissions to the file system folder where you want to create the logs	N/A
vasd daemon	The client computer object is expected to have read access to user and group attributes, which is the default. In order for Authentication Services to update the host object operating system attributes automatically, set the following rights for "SELF" on the client computer object: Write Operating System , Write operatingSystemHotfix , and Write operatingSystemServicePack .	vasd must run as root
QAS/VAS PAM module	N/A (updated by means of vasd)	Any local user
QAS/VAS NSS module vastool nss	N/A (updated by means of vasd)	Any local user

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

Function	Active Directory permissions	Local client permissions
<code>vastool info { site domain domain -n forest-root forest-root -dn server acl }</code>	N/A	Any local user
<code>vastool info { id domains domains -dn adsecurity toconf }</code>	Read permission for the desired objects (regular Active Directory user)	Any local user
<code>vastool isvas</code> <code>vastool inspect</code> <code>vastool license</code>	N/A	Any local user
<code>vastool kinit</code> <code>vastool klist</code> <code>vastool kdestroy</code>	Local client needs permissions to modify the keytab specified; default is the computer object, which is root.	Any local user
<code>vastool ktutil</code>	N/A	root if you are using the default <code>host.keytab</code> file
<code>vastool list</code> (with <code>-l</code> option)	Read permission for the desired objects (regular Active Directory user)	Any local user
<code>vastool load</code>	Permissions to create users and groups in the desired container	Any local user
<code>vastool merge</code> <code>vastool unmerge</code>	N/A	root
<code>vastool passwd</code>	Regular Active Directory user	Any local user
<code>vastool passwd <AD user></code>	Active Directory user with password reset permission	Any local user
<code>vastool schema list</code> <code>vastool schema detect</code>	Regular Active Directory user	Any local user

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

Authentication Services encryption types

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

Table 7: Authentication Services: Encryption types

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

Management Console for Unix requirements

One Identity recommends that you install One Identity Management Console for Unix, a separate One Identity product that provides a management console that is a powerful and easy-to-use tool that dramatically simplifies deployment of Authentication Services agents to your clients. The management console streamlines the overall management of your Unix, Linux, and macOS 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: <ul style="list-style-type: none"> Windows x86 (32-bit) Windows x86-64 (64-bit)

Component	Requirements
	<ul style="list-style-type: none"> • 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	<p>Click www.oneidentity.com/products/authentication-services/ to view a list of Unix, Linux, and Mac platforms that support Authentication Services.</p> <p>Click www.oneidentity.com/products/privilege-manager-for-unix/ to review a list of Unix and Linux platforms that support Privilege Manager for Unix.</p> <p>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.</p> <p>Considerations:</p> <ul style="list-style-type: none"> • To enable the Management Console for Unix server to interact with the host, you must install both an SSH server (that is, <code>sshd</code>) and an SSH client on each managed host. Both OpenSSH 2.5 (and higher) and Tectia SSH 5.0 (and higher) are supported. • Management Console for Unix does not support Security-Enhanced Linux (SELinux) • When you install Authentication Services on Oracle Solaris 11, the Oracle Solaris 10 packages are installed.
Default memory requirement	<p>1024 MB</p> <p>NOTE: See <i>JVM memory tuning suggestions</i> in the <i>One Identity Management Console for Unix Administration Guide</i> for information about changing the default memory allocation setting in the configuration file.</p>

Network requirements

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

Table 9: Network ports

Port	Function
389	Used for LDAP searches against Active Directory Domain Controllers. TCP is normally used, but UDP is used when detecting Active Directory site membership.

Port Function

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 macOS systems to act as full citizens within Active Directory, you must install and configure Authentication Services:

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 management 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 macOS 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 management 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 macOS hosts.

You can install the management 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 Administration Guide* contains detailed instructions for installing the management console on all of these platforms. Use the following procedure to install the console on a supported Windows platform from the Authentication Services 4.2.3 distribution media.

Of course, you can install Authentication Services without using Management Console for Unix. If you wish to do that, skip these instructions and go to [Installing and joining from the Unix command line](#) on page 50.

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 management console on a supported Windows platform

1. Mount the distribution media.

Autorun starts automatically.

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

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

The install wizard guides you through these setup dialogs:

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

Wait until it:

- Extracts and installs Management Console for Unix on your computer.
- Configures the database and service on the server.

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

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

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. In 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 [Customizing installation options](#) on page 26.

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

Customizing installation options

To install specific Authentication Services Windows components

1. From the **Software License Agreement** dialog, click the **Customize installation options** link.
2. In the **Installation Options** dialog, select the components and options you want to install and click **OK**.

Available components:

- Core Components (required)
- ADUC Extensions
- Group Policy Extensions
- Documentation
- Authentication Services Control Center

Available options:

- Install Start Menu Shortcuts
- Install Desktop Shortcuts

| NOTE: You must install the Core Components.

3. To add other Authentication Services components later or modify the current installation of Authentication Services, run the setup wizard again.

From the root of the distribution media, double-click the autorun application.

Installing using `msiexec.exe`

You can install specific Authentication Services components from the Windows command line using `Msiexec.exe`, the Microsoft Windows Installer program, which processes product installation files in the .MSI format. You can either double-click the individual Authentication Services component .msi files or you can run `msiexec.exe` to install, modify, and perform other operations from the Windows command line.

The individual Authentication Services component .msi files, located on the distribution media in the `windows` folder, are:

- `aducX64.msi`: Installs the Active Directory Users and Computers Unix extensions for user and group management on a Windows 64-bit platforms.
- `aducX86.mis`: Installs the Active Directory Users and Computers Unix extensions for user and group management on a Windows 32-bit platforms.
- `cc.msi`: Installs the Control Center extension.
- `corX64.msi`: Installs the core packages on a Windows 64-bit platform.
- `coreX86.msi`: Installs the core packages on a Windows 32-bit platform.

- doc.msi: Installs the User Documentation.
- GpSettingsX86: Installs the Authentication Services Group Policy settings reporting library used by third parties such as Change Auditor and Group Policy Manager to report on Group Policy settings.
- gpX64.msi: Installs the Group Policy extension on a Windows 64-bit platforms.
- gpX86.msi: Installs the Group Policy extension on a Windows 32-bit platforms.

You can use the following properties on the command line when installing the individual Authentication Services components.

Table 10: MSI properties

MSI property	Description
INSTALLFOLDER	Specifies the directory where you want to install the package. (Core X86 only.) Default: %PROGRAMFILES(X86)%\Quest Software\Authentication Services
INSTALLDESKTOPSHORTCUTS	Specifies whether or not to install desktop shortcuts. Default: 0 (Do not install desktop shortcuts.)
INSTALLSTARTMENUSHORTCUTS	Specifies whether or not to install Start menu shortcuts. Default: 0 (Do not install Start menu shortcuts.)
ARPSYSTEMCOMPONENT	Specifies whether or not to add an entry in the Uninstall or change a program interface (Add/Remove Programs) for each individual component (ADUC, Group Policy, and Control Center). Default: 0 (Add entry in Add/Remove Programs.)
NOCHANGEPSPOLICY	Specifies whether or not to allow PowerShell execution policy modifications. (Core X86 only.) Default: 0 (Allow PowerShell policy modifications.)

The following procedures show examples of using the MSI Properties from the Windows command line.

To install Authentication Services Windows components using Msiexec.exe

1. To install the Control Center, enter the following:

```
msiexec /i cc.msi
```

| **NOTE:** Run msiexec -help to see the full command syntax.

2. To specify the install directory path for the core packages, enter:

```
msiexec INSTALLFOLDER=%SystemDrive%\<Directory> /i coreX86.msi
```

NOTE: By default, the installation directory is:

- On Windows 64-bit platforms:

```
%SystemDrive%\Program Files\Quest Software\Authentication Services
```

- On Windows 32-bit platforms:

```
%SystemDrive%\Program Files (x86)\Quest Software\Authentication Services
```

3. To install the Control Center and create a Desktop icon for it, enter:

```
msiexec INSTALLDESKTOPSHORTCUTS=1 /i cc.msi
```

4. To install the Control Center and create a **Start** menu shortcut for it, enter:

```
msiexec INSTALLSTARTMENUSHORTCUTS=1 /i cc.msi
```

5. To install the ADUC extensions and add a separate entry in the **Uninstall or change a program** interface for it, enter:

```
msiexec ARPSYSTEMCOMPONENT=0 /i aducX64.msi
```

NOTE: Setting ARPSYSTEMCOMPONENT to 1 prevents the application from displaying in the **Uninstall or change a program** interface (Add/Remove Programs).

6. You can apply several MSI properties simultaneously, as in the following example:

```
Msiexec.exe INSTALLFOLDER=C:\foo INSTALLDESKTOPSHORTCUTS=1  
INSTALLSTARTMENUSHORTCUTS=0 ARPSYSTEMCOMPONENT=1 NOCHANGEPSPOLICY=1 /i  
corex86.msi
```

If you run this command line, the Core X86 package will be installed into c:\foo, icons will be added to the Desktop, but no **Start** menu shortcut will be added. Furthermore, this package will not be listed in the **Uninstall or change a program** interface (Add/Remove Programs) and the PowerShell Execution Policy will not be updated.

To uninstall Authentication Services components from the Windows command line

1. To uninstall the Control Center, enter the following:

```
msiexec /uninstall cc.msi
```

NOTES:

You can specify either `/uninstall` or `/x`.

If you manually install MSI files, take care to uninstall them in the reverse order that they are installed. For example if you install CoreX86 and AducExtensionsx86 remove them in this order: AducExtensionsx86, then Corex86.

Configure Active Directory for Authentication Services

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

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

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

For more information, see [Version 3 Compatibility Mode](#) on page 33.

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 management 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](#) on page 57.

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 functionality.

NOTE: If you do not configure Active Directory for Authentication Services, you can run your Authentication Services client agent in Version 3 Compatibility Mode, which allows

you to join a host to an Active Directory domain.

For more information, see [Version 3 Compatibility Mode](#) on page 33.

To configure Active Directory for Authentication Services

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

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

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

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

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

- c. Optionally enter the domain or domain controller and click **Next**.

3. In the **License Authentication Services** dialog, browse to select your license file and click **Next**.

Refer to [About Authentication Services licenses](#) on page 10 for more information about licensing requirements.

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

4. In the **Configure Settings in Active Directory** dialog, accept the default location in which to store the configuration or browse to select the Active Directory location where you want to create the container and click **Setup**.

NOTE: You must have rights to create and delete all child objects in the selected location. For more information on the structure and rights required see [Windows permissions](#) on page 12.

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

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

Proceed to [Configure Unix agent components](#) on page 33

About Active Directory configuration

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

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

For more information, see [Version 3 Compatibility Mode](#) on page 33.

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

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

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

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

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

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

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

For more information, see [Creating the application configuration from the Unix command line](#) on page 57.

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

showInAdvancedViewOnly for the Active Directory configuration root container and all child objects.

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

The following table summarizes the required rights.

Table 11: 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, GSSAPI based authentication-only environments, or configurations where the Authentication Services agent will auto-generate posix attributes for Active Directory Users and Groups objects.

Version 3 Compatibility Mode

When upgrading to or installing Authentication Services 4.x, you can choose not to configure Active Directory for Authentication Services and run your Authentication Services client agent in Version 3 Compatibility Mode. While this prevents you from running the Control Center and accessing its many features and tools, you can join a host to an Active Directory domain when operating in Version 3 Compatibility Mode.

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

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

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

Best practice

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

There are two ways to create the application configuration:

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

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. If you wish to do that, skip these instructions and go to [Installing and joining from the Unix command line](#) on page 50.

To start the mangement console

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

Set up Management Console for Unix wizard

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

Choose one of these options:

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

This option allows you to use the core features of the console and limits access to the console to the default **supervisor** account only.

- **Walk me through the configuration steps for using AD user accounts for logon to the console**

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

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

Choose an option and click **Next**.

NOTE: If you choose the **Skip** option, the **Identify Console** dialog displays. For more information, see [Identify Console dialog](#) on page 36.

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 in to the console as **supervisor** and configure the console for Active Directory from **System Settings**. See *Active Directory Configuration* in the *One Identity Management Console for Unix Administration Guide* for more information.

Configure Console for Active Directory Logon dialog

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

To configure the management console for Active Directory logon

1. In the **Configure Console for Active Directory Logon** dialog, enter a valid Active Directory domain in the forest, in the form **example.com**.
2. Enter the credentials for an Active Directory account that has logon rights.
Enter a sAMAccountName, which uses the default domain or a User Principal Name, as in **username@domain**. The wizard uses these credentials to configure the management console for use with 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**.

| **NOTE:** This is a read-only operation; no changes are made to Active Directory.

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

Set up console access by role dialog

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

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

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

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

By default the management console assigns users to **All Roles**, which gives those accounts permissions to access and perform all tasks within the console. See *Console Roles and Permissions System Settings* in the *One Identity Management Console for Unix Administration Guide* for details.

3. Click in the **Roles** cell to activate a drop-down menu from which you can choose a role for the user account.

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

4. Click **Next** to save your selections.

The **Identify Console** dialog opens.

Identify Console dialog

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

To identify the management console

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

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

Set Supervisor Password dialog

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

To set the supervisor password

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

The **Summary** dialog displays.

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

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

Summary dialog

To complete the Management Console for Unix Setup wizard

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

The Management Console for Unix login screen opens.

Logging in to Management Console for Unix

Whenever you launch the management 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 management 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 management console, that is, defined during the post-installation configuration. See *Setup Console Access by Role* in the online help for details. To add additional accounts to this access list, see *Add (or Remove) Role Members* in the online help for details.

To log on to the management 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 management 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 redisplay, allowing you to enter a different account.

Prepare Unix hosts

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

Adding hosts to the management console

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

To add hosts to the management 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 management console and either click the **Add** button or press **Enter**.

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

3. To add hosts from a `known_hosts` file, click the **Import** button.
 - a. In the **Import hosts from file** dialog, browse to select a `.txt` file containing a list of hosts to import.

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

NOTE: The valid format for an import file is:

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

See *Known_hosts File Format* in the online help for more information about the supported `known_hosts` file format.

4. Once you have a list of one or more hosts to add, if you do not wish to profile the hosts at this time, clear the **Profile hosts after adding** option.

NOTE: If you add more hosts to the list than selected in the **Rows to show** drop-down menu in the **View** panel of the tool bar, this option is disabled.

5. If you do not clear the **Profile hosts after adding** option in the **Add Hosts** dialog, when you click **OK**, the **Profile Host** dialog prompts you to enter the user credentials to access the hosts. Refer to [Profiling hosts](#) on page 39, which walks you through the host profile steps.
6. If you clear the **Profile hosts after adding** option in the **Add Hosts** dialog, when you click **OK**, the **Add Hosts** dialog closes and control returns to the management console.

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

Profiling hosts

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

To profile hosts

1. Select one or more hosts in the **All Hosts** view and click **Profile** from the **Prepare** panel of the tool bar, or open the **Profile** menu and choose **Profile**.
2. In the **Profile Host** dialog, enter user credentials to access the hosts.
If you selected multiple hosts, you are asked if you want to use the same credentials for all the hosts (default) or enter different credentials for each host.
3. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter the following information:
 - a. Enter the user name and password to log onto the selected hosts.
 - b. Optionally, enter the SSH port to use. It uses port 22 by default.
 - c. To save the credentials entered for the host, select the **Save my credentials on the server** option.

Once saved, the management 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 management console needs credentials to complete a task on the host during a logon session. Once entered, the management 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 management console encrypts the credentials and saves them in the Java keystore. Saved user names and passwords persist across logon sessions, and when needed, the management 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 management console to prompt you to review and accept new SSH keys for the selected hosts (which do not have previously cached SSH keys), clear the **Automatically accept SSH keys** option before you click **OK**.

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

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

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

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

Configuring automatic profiling

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


BEST PRACTICE: Configure newly added hosts for auto-profiling before you perform any other actions so that the management console dynamically updates user and group

information. See *UID or GID Conflicts* in the online help.

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

The cron job detects changes to the following:

- Local users, groups, or shells
- Installed Authentication Services or Privilege Manager software
- Authentication Services access control lists
- Authentication Services mapped user information
- Privilege Manager configuration
- Authentication Services configuration
- Privilege Manager licenses

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

To configure automatic profiling

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

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

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

- **Create a user service account on the host**

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

1. Creates "questusr," the user service account, and a corresponding "questgrp" group on the host that the management console uses for automatic profiling.
2. Adds *questusr* as an implicit member of *questgrp*.

-OR-

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

Click **Select** to browse for a user.

4. Click **OK** in the **Profile Automatically** dialog.

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

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

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

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

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

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

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

To re-create the "questusr" account

- Re-profile the host.
 - Reconfigure the host for automatic profiling.
- In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

NOTE: This task requires elevated credentials.

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

- a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.
- b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid is displayed that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

To disable automatic profiling

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

When you disable auto-profiling for a host, the management 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.

Installing software on hosts

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

To install Authentication Services software on hosts

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

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

The tool bar button will not be active if:

- You have not selected any hosts.
 - You have selected multiple hosts with different states (added, profiled, or joined).
2. In the **Install Software** dialog, select the Authentication Services software products you want to install and click **OK**.

- **Authentication Services Agent (Required):** Select to allow Active Directory users access to selected host. Authentication Services provides centralized user and authentication management. It uses Kerberos and LDAP to provide secure data transport and an authentication framework that works with Microsoft Active Directory. Components include `vasd`, `nss_vas`, `pam_vas`, and `vastool`.
- **Authentication Services for Group Policy (Required):** Select to install the Group Policy component that provides Active Directory Group Policy support for Unix, Linux, and macOS platforms.
- **Authentication Services for NIS:** Select to install the NIS Proxy component that provides the NIS compatibility features for Authentication Services. `vasyp` is a NIS daemon that acts as a `ypserv` replacement on each host.
- **Authentication Services for LDAP:** Select to install the LDAP Proxy component that provides a way for applications that use LDAP bind to authenticate users to Active Directory without using secure LDAP (LDAPS). Instead of sending LDAP traffic directly to Active Directory domain controllers, you can configure applications to send plain text LDAP traffic to `vasldapd` by means of the loopback interface. `vasldapd` proxies these requests to Active Directory using Kerberos as the security mechanism.
- **Dynamic DNS Updater:** Select to install the Dynamic DNS Updater component that provides a way to dynamically update host records in DNS and can be triggered by DHCP updates.
- **Defender PAM Module:** Select to install the Defender authentication components for PAM based Unix/Linux systems. Includes PAM module, documentation, and utilities to appropriately configure the PAM subsystem for Active Directory/Defender OTP authentication.

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

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

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

NOTE: This task requires elevated credentials.

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

- a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.
- b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

Checking readiness

Once you install the software on your remote hosts, the management 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 management console online help for more information.

To check hosts for Active Directory Readiness

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

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

- a. If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.
- b. If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays that allows you to enter different credentials for each host listed. Place your cursor in a cell in the grid to activate it and enter the data.

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

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

NOTE: This task requires elevated credentials. The management 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.

Check QAS agent status

You can either check the health status of Authentication Services agents manually, or you can configure the management console to automatically check the QAS Agent Status and report any warnings or failures to the console.

NOTE: Running the *Check QAS Agent Status* commands requires:

- You are logged on as an Active Directory account in the **Manage Hosts** role.
- The hosts have Authentication Services 4.0.3.78 (or later) Agent software installed.

See *Check QAS Agent Status Commands Not Available* in the management console online help for more information.

Manually checking QAS agent status

To check QAS agent status

1. Select one or more hosts on the **All Hosts** view, open the **Check** menu from the **Prepare** panel of the tool bar and choose **Check QAS agent status**.
2. In the **Log on to Host** dialog, enter the user credentials to access the selected hosts and click **OK**.

A progress bar displays in the **Task Progress** pane and the **Host Notifications** tab indicates the number of hosts with warnings or failures detected.

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.

- If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.
 - If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays allowing 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.
3. Select the **Host Notifications** tab to view the reported warnings or failures.
See *View the QAS Agent Status* in the management console online help for details.

Configuring to automatically check QAS agent status

To have updated information about the status of Authentication Services agents, you can configure the management console to periodically check the QAS Agent Status automatically. If it detects a status change on the host, it reports the following warnings or failures to the Host Notifications tab:

- Critical Failure
- Failure
- Warning

To configure the console to automatically check the QAS agent status

1. Select one or more hosts on the **All Hosts** view, open the **Check** menu from the **Prepare** panel of the tool bar, and choose **Check QAS Agent Status**

automatically.



NOTE: This option is only available for multiple hosts if all hosts are in the same "Check QAS Agent Status" state; that is, they all have automatic status checking turned on, or they all have automatic status checking turned off.

2. Select the **Check status automatically** option, set the frequency for the health status check, and click **OK**.

NOTE: Use standard crontab syntax when entering Advanced schedule settings.

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

NOTE: This task requires elevated credentials.

When configured for automatic checking, the **Authentication Services state** column on the **All Hosts** view displays the  icon. Then, if the server does not receive a heartbeat in over four hours (by default), it displays the  icon. No icon in the **Authentication Services state** column indicates the host is not configured to check the QAS agent status automatically.

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.

- If you selected multiple hosts and the **Use the same credentials for all selected hosts** option, enter your credentials to log on to access the selected hosts and click **OK**.
- If you selected multiple hosts and the **Enter different credentials for each selected host** option, a grid displays allowing 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.

NOTE: If you receive a GID conflict error, see *UID or GID Conflicts* in the online help.

4. View the QAS Agent Status for each host on the **Host Notification** tab.

See *View the Authentication Services Status Errors* in online help for details.

When you configure a host to check the QAS agent status automatically, the management console,

- a. Creates "questusr" (the service account user), if it does not already exist, and, a corresponding "questgrp" group on the host that the management console uses for automatic QAS agent status checking.
- b. Adds *questusr* as an implicit member of *questgrp*.
- c. Adds the auto-check SSH key to *questusr*'s *authorized_keys*, `/var/opt/quest/home/questusr/.ssh/authorized_keys`.
- d. Verifies the service account user can log in to the host.
- e. Creates a cron job that runs QAS agent status according to the specified interval.

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 users and groups in a read-only fashion; however, the management console does not use the *questusr* account to make changes to any configuration files.

NOTE: If *questusr* is inadvertently deleted from the console, the console will not be updated. To recreate the "questusr" account, re-configure the host for automatic QAS agent status checking.

To disable automatic status checking

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

When you disable auto-status checking for a host, the management console

1. Leaves the "questusr" and the corresponding "questgrp" accounts on the host.
2. Leaves *questusr* as an implicit member of *questgrp*.
3. Removes the auto-check SSH key from that user's `authorized_keys` file.
4. Removes the cron job on the host.

Installing and joining from the Unix command line

While you can use Management Console for Unix to install and configure Authentication Services as explained in [Installing and configuring Authentication Services](#) on page 23, you can also manually install the Authentication Services agent on each Unix, Linux, or macOS host from the command line.

This section walks you through the process of installing the Authentication Services Unix agent directly from the command line. For information about installing, upgrading, and uninstalling the Authentication Services agent on supported platforms in an enterprise environment using platform package management tools, refer to [Enterprise package deployment](#) on page 107.

Before installing and configuring the Authentication Services Unix agent, One Identity recommends that you run the `preflight` tool to check a host's suitability to run Authentication Services. After you determine that the Unix host is ready, run the Authentication Services installation script, `install.sh`, to install the Unix/Linux agent.

The Authentication Services pre-installation diagnostic tool

One Identity provides the `preflight` utility to check a host's suitability to run Authentication Services by verifying a number of environmental considerations necessary for joining an Active Directory domain.

This utility obtains answers to the following questions:

- Does Authentication Services support the host on which this utility is being run?
- Are the operating system and any patches at requisite levels?
- Is there at least one visible domain controller (DC)?
- Are global catalogs available on any of the domain controllers?
- Are all services needed by Authentication Services available?
- Is an Authentication Services application configuration set up on the target domain?

The `preflight` command-line utility performs the following verifications.

Install checks:

- Check for supported operating system and correct operating system patches.
- Check for sufficient disk space to install Authentication Services.

Join checks:

- Check that the hostname of the system is not `localhost`.
- Check if the name service is configured to use DNS.
- Check `resolv.conf` for proper formatting of name service entries and that the host can be resolved.
- Check for a name server that has the appropriate DNS SRV records for Active Directory.
- Detect a writable domain controller with UDP port 389 open.
- Detect Active Directory site, if available.
- Check if TCP port 464 is open for Kerberos `kpasswd`.
- Check if UDP port 88 and TCP port 88 are open for Kerberos traffic.
- Check if TCP port 389 is open for LDAP.
- Check for a global catalog server and if TCP port 3268 is open for communication with global catalog servers.
- Check for a valid time skew against Active Directory.
- Check for the Authentication Services application configuration in Active Directory.

Post-join checks:

- Check if TCP port 445 is open for Microsoft CIFS traffic.

You can find the `preflight.sh` script at the root of the ISO. This script runs the correct `preflight` version for your system.

The most important options and arguments to `preflight` are:

- **`domain-name`**

The domain you want to join with Authentication Services.

- **`-u username`**

An identity with administrator rights for the Active Directory domain you want to join.

NOTE: The `preflight` utility does not make any changes to your system.

Running preflight

To run preflight

1. Mount the Authentication Services distribution media.
2. Enter the following command at the root of the Authentication Services ISO:

```
# ./preflight.sh -u Administrator example.com
```

where Administrator is your user name and example.com is the name of your domain.

By default, preflight outputs the results of the verifications for the three types of checks (Install checks, Join checks, and Post-join checks) to the console. Run the preflight utility with the `--verbose` option to obtain detailed information about the various checks in those categories.

The last line of the output tells you whether you are ready to continue deploying Authentication Services.

If you did not get a Preflight Checks complete with status Success message, correct any failures indicated before continuing with the Authentication Services installation. Be aware of any "Advisories" that it returns, as they may effect your ability to install or join.

NOTE: If you get a message that says, Unable to locate Authentication Services Application Configuration, you can ignore that error for now and proceed with the Authentication Services installation. The Authentication Services Active Directory Configuration Wizard starts automatically to help you configure Active Directory for Authentication Services the first time you start the Control Center. Or, you can create the Authentication Services application configuration from the command line, as explained in [Creating the application configuration from the Unix command line](#) on page 57.

NOTE: For information about other preflight options, either run `preflight --help` or refer to the *preflight man page* located in the docs directory of the installation media. See [Resolving preflight failures](#) on page 100 for additional help in resolving issues.

The Authentication Services install script

Follow the steps in this topic if you are installing a Authentication Services 4.2.3 for the first time; that is, if you are not upgrading from VAS 3.5.

The Authentication Services installation script, `install.sh`, installs Authentication Services, joins the domain, and allows you to install licenses. You can run the install script in interactive mode by using the `-i` option. This provides you with a menu of valid operations to perform, including **Running preflight**.

You can also automate the installation process by running `install.sh` in "unattended" mode using `-q` option. In this mode you may specify a set of commands for the script to perform.

NOTE: For more information on the Authentication Services installation script, run:

```
install.sh --help
```

Installing the Authentication Services agent

To install the Authentication Services agent with the installation script

1. Log in and open a root shell.
2. Mount the installation ISO for your selected platform and navigate to the mount point.
3. Run `install.sh` by entering the following command:

```
# ./install.sh vasclnt
```

NOTE: See [Installing the Authentication Services agent package](#) on page 107 for a list of the Authentication Services Agent installation commands.

After installing Authentication Services some services such as `cron`, `sshd`, and `gdm` may need to be restarted in order to reload NSS configuration. If you are unsure of which services to restart, reboot the system.

Installation script options

If you run `install.sh` with no option, it installs (or upgrades) Authentication Services and Authentication Services Group Policy, installs the license, and joins the domain.

The following is a list of the available options to the Authentication Services install script.

Table 12: install.sh: Options

Option	Function
-d	Turn on debug.
-h	Help; displays usage information including a brief summary of options.
--help	Displays full script help.
-v, --version	Displays version and lists products available on this ISO.
-l path	License; path to One Identity license file to copy (unattended mode). Not valid with -i (interactive mode).
-p	Specify alternate ISO path to search for install packages.

Option	Function
-t	Test host and ISO and report on what is installed and available.
<none>	Simple mode.
-i	Interactive mode; provides a menu showing choices based on existing Authentication Services software installation and includes a help mode.
-q	Unattended mode; executes script in unattended (automatic) mode; requires other options.
-a	Accept License; signals acceptance of One Identity LLC EULA.

Table 13: Special commands

Command	Description
upgrade	Upgrades all products on the system.
remove	Removes all products from the system.
join	Executes interactive <code>vasjoin.sh</code> script. Not valid with <code>-q</code> (Unattended mode).
preflight	Executes interactive preflight test. Not valid with <code>-q</code> (Unattended mode).
license	Executes interactive install of license files (or use <code>-l</code> option). Not valid with <code>-q</code> (Unattended mode).

In unattended mode, the following arguments are useful for scripting the components you want to install or uninstall.

Table 14: `install.sh`: Unattended mode arguments

Argument	Function
vascert	Installs or upgrades Authentication Services Certificate Autoenrollment.
vasclnt	Installs or upgrades Authentication Services agent.
vasdev	Installs or upgrades Authentication Services SDK.
vasgp	Installs or upgrades Authentication Services Group Policy agent.
vasproxy	Installs or upgrades Authentication Services Proxy daemon.
vassc	Installs or upgrades Authentication Services for Smart Cards agent.
vasyp	Installs or upgrades Authentication Services YP server.
novascert	Uninstalls Authentication Services Certificate Autoenrollment.
novasclnt	Uninstalls the Authentication Services agent.
novasdev	Uninstalls the Authentication Services SDK.

Argument	Function
novasgp	Uninstalls the Authentication Services Group Policy agent.
novasproxy	Uninstalls the Authentication Services Proxy daemon.
novassc	Uninstalls the Authentication Services for Smart Cards agent.
novasyp	Uninstalls the Authentication Services YP server.

Licensing Authentication Services

You must have the Authentication Services license installed for full Authentication Services functionality on Unix.

There are four ways to manage licenses:

- **Using the Control Center**

One Identity recommends this as a best practice. For more information, see [Adding licenses using the Control Center](#) on page 66.

- **Using the Authentication Services Group Policy utilities**

For more information, see *Licensing Policy* in the *Authentication Services Administration Guide*.

- **Running the install.sh script with the -l option**

This allows you to enter a path. The script then places the license in the proper location. For more information, see [Installation script options](#) on page 53.

- **Installing Licenses from the command line**

For more information, see [Installing licenses from the command line](#) on page 56.

To obtain a license, use the [Licensing Assistance](#) page on the One Identity support page or contact your account representative.

NOTE: If you are running Management Console for Unix with a licensed version of Authentication Services, any time you make a change to the Authentication Services licensing, go into the management console **System Setting | Licenses** and click the **Check for licenses** button to refresh the product license information in Management Console for Unix.

Verifying Authentication Services license information

To verify that you have a valid Authentication Services license

1. Run the following `vastool` command:

```
vastool license -q
```

You will see output similar to the following if you have a valid license installed:

```
Number of Unix Enabled users in use:    150
---QAS---
Number of Licensed Unix Enabled Users: 1000
Valid licenses:                        1
```

Installing licenses from the command line

With root privileges, you can manually install a valid license by copying the new license file to the `licenses` directory on the Unix host.

To install a Authentication Services license manually

1. Copy the license file to the `/etc/opt/quest/vas/.licenses` directory.
2. Ensure the permissions on the license file are set to `0644`.
3. Restart `vasd` as root by running the command corresponding to your platform:

- **Linux/Oracle Solaris:**

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

- **HPUX:**

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

- **AIX:**

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

- **macOS:**

```
launchctl unload /Library/LaunchDaemons/com.quest.vasd.plist
launchctl load /Library/LaunchDaemons/com.quest.vasd.plist
```


Creating the application configuration from the Unix command line

Before you join a Unix client to an Active Directory domain, One Identity recommends that you create the application configuration in the domain to which you are joining to utilize full Authentication Services 4.2.3 functionality. While the Authentication Services Active Directory Configuration Wizard starts automatically to help you configure Active Directory for Authentication Services the first time you start the Control Center, you do not need to have a Windows console to create the application configuration. You can run the `vastool configure ad` command from the Unix command line to create it. This is typically a one-time process.

NOTE: You only need to create one Authentication Services application configuration per forest. For more information, see [Version 3 Compatibility Mode](#) on page 33.

To create the Authentication Services application configuration

1. Run the following command from the Unix command line:

```
# /opt/quest/bin/vastool ad -u <user> configure -d <domain>
```

By default, Authentication Services creates the application configuration in the Program Data container; however, if you do not have rights to create an organizational unit in the Program Data container, you can create the Authentication Services application configuration in any location you have rights to by specifying the DN (distinguished name) of the creation location, as follows:

```
vastool -u <user> configure -d <domain> ou cn=myou,dc=example,dc=com
```

2. Enter the user's password when prompted.

Changing the schema configuration mode

When you create the Authentication Services application configuration, you set the global schema configuration mode to R2 by default. However, you can optionally configure Authentication Services for "schemaless" operation using the `schema configure` command.

To switch to a schemaless configuration

1. Run the following command:

```
# /opt/quest/bin/vastool -u <user> schema -d <domain> configure schemaless
```

The `schema configure` command only allows you to set the schema mode to either R2 or "schemaless" modes. To set the schema configuration to any other mode, you must do so from the Control Center **Preferences**.

2. Enter the user's password when prompted.

Joining the domain

For full Authentication Services functionality on Unix, you must join the Unix system on which you installed the Authentication Services agent to the Active Directory domain. You can join an Active Directory domain either by running `vastool join` from the command line or the interactive join script, `vasjoin.sh`.

Before you join the Unix host to the Active Directory domain, you may want to determine if you are already joined.

To determine if you are joined to an Active Directory domain

1. Run the following command:

```
# /opt/quest/bin/vastool info domain
```

If you are joined to a valid domain this command returns the domain name. If you are not joined to a domain, you will see the following error:

```
ERROR: No domain could be found.  
ERROR: VAS_ERR_CONFIG: at ctx.c:414 in _ctx_init_default_realm  
default_realm not configured in vas.conf. Computer may not be joined to domain
```

Joining the domain using VASTOOL

You can join your Unix host to Active Directory with the `vastool join` command directly from the command line.

Before you join the Authentication Services agent to the Active Directory domain, collect the following information:

- The DNS name of the Active Directory domain of which you want the Authentication Services agent to be a member.
- The user name and password of a user that has sufficient administrative privileges to create computer objects in Active Directory.

To join Active Directory using vastool join

1. Run the following command as the root user at a shell prompt:

```
# /opt/quest/bin/vastool -u <user> join <domain-name>
```

2. Enter the user's password when prompted.

The vastool join results are shown on the shell's standard output.

NOTE: vastool join supports many options that allow you to customize the way the computer is joined to the domain. You can specify the name of the computer object. You can join to a specific organizational unit or use a pre-created computer object. For a list of all vastool join options, refer to the *vastool man page*.

Joining the domain using VASJOIN script

Rather than using the vastool join command from the command line, you can join your Unix host to Active Directory using the interactive join script, *vasjoin.sh*. The script walks you through the domain join process, calling the vastool join command.

The *vasjoin.sh* script is in */opt/quest/libexec/vas/scripts/* directory. You can use most of the standard vastool join command options when running it. However, you can run the join script with no options; it only requires that you supply the domain name and the name of a user with sufficient Active Directory privileges to perform the join.

Table 15: Common vasjoin script options

Option	Function
-h	Help; displays options including how to pass vastool join options.
-q	Unattended or "quiet" mode; displays less verbose: no explanations, asks no questions.
-i	Interactive mode; prompts for common options.
<none>	Simple mode; installs vasclnt and vasgp with options to add license and join domain.

To join Active Directory using the vasjoin script

1. Run the script as the root user at a shell prompt, as follows:

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

The script ensures that your local host's time is synchronized with that of the controller in the domain you want to join (in order to satisfy Kerberos), then performs the join for you by running vastool join as follows:

```
vastool -u <username> join <domain-name>
```

2. Follow the prompts to complete the join process.

NOTE: Run the script in interactive mode as follows:

```
/opt/quest/libexec/vas/scripts/vasjoin.sh -i
```

In interactive mode, it prompts you for specific information and allows you to either save the resulting `vastool join` command in a script or execute the command immediately.

The script presents defaults as part of the prompting and, if you accept them all, the result is identical to running the script in simple mode.

The information gathered by the full, interactive mode of `vasjoin.sh` includes the following:

- Specific domain controllers to use
- Domain to join
- User, usually administrator, to use in joining
- Keytab file
- Confirm fixing of Kerberos clock skew, if any
- Overwrite your host's existing Active Directory ComputerName object
- Change the name of the AD ComputerName object
- AD container in which to put the ComputerName object
- Site name
- UPM mode (yes or no)
- User search path on which to look for Active Directory users
- Alternate group search path
- Workstation mode (yes or no)
- Alternate domains in which to search if you want cross-domain logins
- Self-enrollment of existing `/etc/passwd` users (yes or no)
- Shows path to `lastjoin` (`/etc/opt/quest/vas/lastjoin`)

The `lastjoin` file contains something similar to:

```
/opt/quest/bin/vastool -u administrator join -f acme.com
```

Dynamic DNS update tool

When Authentication Services joins a new computer to a domain, it becomes known to the LDAP and Kerberos protocols, but not to DNS. This is because the IP address of the host is

not directly under the control of this part of Active Directory.

Although Active Directory comes with a integrated DHCP and DNS servers, some sites run their own DHCP servers. This means that the leased IP addresses must be communicated to Active Directory's DNS server through another (often manual) means.

The One Identity Dynamic DNS Update Tool, `dnsupdate`, performs this communication. It can automatically and securely inform Active Directory's DNS server of any host IP address changes due to DHCP lease acquisition and renewal.

Because `dnsupdate` uses Kerberos to authenticate itself to the DNS server, only the computer joined with that name can update its record.

When you run the Authentication Services installation script, `install.sh`, in interactive mode (the `-i` option), it gives you an option to install the One Identity Dynamic DNS Update Tool. Dynamic DNS automatically integrates into the host's native DHCP client infrastructure to securely update DNS servers when its IP address changes. For more information about running the `install.sh` script, see [Installation script options](#) on page 53.

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

Getting started with Authentication Services

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

Getting acquainted with the Control Center

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

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

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

Table 16: Control Center: Navigation links

Control Center pane	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 management console within the Control Center or you can run it separately in a supported web browser. The management console is a separate install on Windows, Unix, Linux, or macOS that you can launch from the ISO.

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

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

Management console

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

With the management 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 management console on supported Unix, Linux, and macOS platforms. Once installed, you can access it from a browser using default port of 9443 or from the Control Center.

Group Policy

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

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

Filtering the list of GPOs

To filter the list of GPOs

1. Open the Control Center and click **Group Policy** on the left navigation pane.
2. Expand the **Filter Options** section.
3. Enter all or part of a name to filter the list of GPOs.
4. Open the **Domain** drop-down menu to choose a domain.
5. 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 macOS display.

Editing a GPO

To edit a group policy object

1. Open the Control Center and click **Group Policy** on the left navigation pane.
2. From the **Group Policy** window, select a GPO in the list and click **Actions | Edit GPO**.

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

NOTE: For more information about the group policies, refer to the *Authentication Services Administration Guide*, which can be found on the [Authentication Services - Technical Documentation](#) page of the One Identity support site.

Generating a settings report

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

To generate a settings report

1. Open the Control Center and click **Group Policy** on the left navigation pane.
2. From the **Group Policy** window, select a GPO Name and click **Actions | Settings Report**.

An HTML report of the currently configured Unix and macOS settings displays.

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

Showing files

To open the Windows Explorer

1. Open the Control Center and click **Group Policy** on the left navigation pane.
2. From the **Group Policy** window, select a GPO in the list and click **Actions | Show Files**.

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

Launching GPMC

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

To launch the Group Policy Management Console

1. Open the Control Center and click **Group Policy** on the left navigation pane.
2. From the **Group Policy** window, click **Actions | Launch GPMC**.

Tools

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

- **Authentication Services**
Direct links to installed applications and tools related to Authentication Services.
- **Additional One Identity Products**
Direct links to other One Identity product plugins.

| **NOTE:** The **Additional One Identity Products** link is only available if you have

installed other One Identity products such as Defender, Authentication Services for Smart Cards, or One Identity Active Roles.

- **Other Tools**

Direct links to tools related to Authentication Services.

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

- **Documentation**

Direct links to Authentication Services documentation.

Preferences

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

- [Licensing](#)
- [Global Unix Options](#)
- [Logging Options](#)
- [Custom Unix Attributes](#)

Licensing

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

Refer to [About Authentication Services licenses](#) on page 10 for more information about licensing requirements.

Adding licenses using the Control Center

To add licenses using the Control Center

1. Open the Control Center and click **Preferences** on the left navigation pane.
2. Expand the **Licensing** section.
The list box displays all licenses currently installed in Active Directory.
3. Click **Actions | Add a license**.

4. Browse for the license file and click **Open**.

The license appears in the list box.

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

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

Global Unix Options

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

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

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

Table 17: Unix user defaults

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

Option	Description
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 18: Unix group defaults

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

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

Table 19: Unique IDs

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

Option	Description
	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.

Enabling debug logging on Windows

To enable debug logging for all Authentication Services Windows components

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

Debug generates the most log output. Higher levels generate less output. You can set the **Log level** to **Disabled** to disable logging.

4. Click  to specify a folder location where you want to write the log files.

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

Custom Unix Attributes

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

Authentication Services supports the following schema templates if the required schema is installed:

Table 20: Unix schema attributes

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

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

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

Active Directory schema extensions

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

Table 21: Active Directory schema extensions

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

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

Configuring a custom schema mapping

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

To configure a custom schema mapping

1. Open the Control Center and click **Preferences** on the left navigation pane.
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 71.

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. For more information, see [Prepare Unix hosts](#) on page 37.

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 management console.

Adding a local group

You can use the management 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, open the **Host | All Hosts** view.
2. From the **All Hosts** view, double-click a host name to open its properties.
3. Select the **Groups** tab and click **Add Group**.
4. In the **Add New Group** dialog, enter **localgroup** as a local group name in the **Group Name** box and click **Add Group**.
5. 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 management console.

Adding a local user account

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

To add a local user account

1. From the Management Console for Unix, open the **Host | All Hosts** view.
2. From the **All Hosts** view, double-click a host name to open its properties.
3. Select the **Users** tab from the host properties and click **Add User**.
4. In the **Add New User** dialog:
 - a. Enter **localuser** as a new local user name in the **Name** box.
 - b. Click **Select Group** browse button next to the **GID** box, to find and select the local group account you set up in [Adding a local group](#) on page 72.
You can also use 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.
5. In the **Log on to Host** dialog, enter your credentials to log in to the host and click **OK**.

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

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

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

Adding an Active Directory group account

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

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, click **Tools** on the left navigation pane.
2. From the **Tools** window, click the **Authentication Services Extensions for Active Directory Users and Computers** link.

The **Active Directory Users and Computers** Console opens.
For Windows 7, you must have the Remote Server Administration Tools installed and enabled.

3. Expand the **domain** folder and select the **Users** folder.
4. Click the **New Group** icon button.
The **New Object - Group** dialog opens.
5. Enter **UNIXusers** in the **Group name** box and click **OK**.

Adding an Active Directory user account

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

To create an Active Directory user account

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

The **New Object - User** wizard guides you through the user setup process.

3. When you enter a password, clear the **User must change password at next logon** option, before you click **Next**.
4. Click **Finish**.
5. Close **Active Directory Users and Computers** and return to the management console.

Changing the default Unix attributes

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

To change the default Unix attributes

1. Open the Control Center and click **Preferences** on the left navigation pane.
2. Expand **Global Unix Options**.
The window displays the current settings for Unix-enabling users, groups and the method used for creating unique IDs.
3. Click **Modify Global Unix Options** on the right side of the window.
The **Modify Global Options** dialog opens.
4. Change the **Login Shell** to **/bin/bash** and click **OK**.
The defaults are saved to Active Directory.

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

Active Directory account administration

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


Enabling local user for AD authentication

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

To enable a local user for Active Directory authentication

1. From the management console, open the **Host | All Hosts** view.
2. From the **All Hosts** view, double-click a host to open its properties.
3. Select the **Users** tab and double-click the **localuser** account to open its properties.

NOTE: To set up this local user account, see [Adding a local user account](#) on page 73.

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



NOTE: To set up this Active Directory user, see [Adding an Active Directory user account](#) on page 74.

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

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

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

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

1. From the **All Local Users** tab, select one or more local Unix users.
2. In the **Require AD Logon** pane, click the  **Search** button to populate the list of Active Directory users.
(Click the  **Directory** button to search in a specific folder.)
3. Select an Active Directory user and click the **Require AD Logon to Host** button at the bottom of the **Require AD Logon** pane.
4. In the **Log on to Host** dialog, verify your credentials to log in to the host and click **OK**.

NOTE: This task requires elevated credentials.

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

Testing the mapped user login

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

To test the mapped user login

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

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

2. If the **PuTTY Security Alert** dialog opens, click **Yes** to accept the new key.

3. Enter the password for **ADuser**, the Active Directory user account you mapped to **localuser**, when you selected the **Require an AD Password to logon to Host** option on the user's properties.
4. At the command line prompt, enter `id` to view the Unix account information.
5. Enter `/opt/quest/bin/vastool klist` to see the credentials of the Active Directory user account.
6. Enter `exit` to close the command shell.

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


Unix-enabling an Active Directory group

To Unix-enable an Active Directory group

1. On the management console's **Active Directory** tab, open the **Find** box drop-down menu and choose **Groups**.
2. Enter a group name, such as **UNIX**, in the **Search by name** box and press **Enter**.
3. Double-click the group name, such as **UNIXusers**, to open its properties.
NOTE: To set up this Active Directory user account, see [Adding an Active Directory group account](#) on page 74.
4. On the **Unix Account** tab, select the **Unix-enabled** option and click **OK**.

Unix-enabling an Active Directory user

To Unix-enable an Active Directory user

1. On the management console's **Active Directory** tab, open the **Find** box drop-down menu and choose **Users**.
2. Click  next to the **Search by name** box to search for all Active Directory users. Or, enter a portion of your **ADuser** logon name in the **Search by name** box and press **Enter**.
3. Double-click **ADuser**, the Active Directory user name, to open its properties.
4. On the **Unix Account** tab, select the **Unix-enabled** option.
It populates the properties with default Unix attribute values.
5. Make other modifications to these settings, if necessary, and click **OK** to Unix-enable the user.

NOTE: There are additional settings that you can set using PowerShell which allows you to validate entries for the GECOS, Home Directory, and Login Shell attributes. For more information, see [Use Authentication Services PowerShell](#) on page 91.

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

Testing the Active Directory user login

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

To test the Active Directory login

1. From the Control Center, under **Login to remote host**, enter:
 - **Host name:** The Unix host name.
 - **User name:** The Active Directory user name, such as **ADuser**.Click **Login** to log in to the Unix host with your Active Directory user account.
2. Enter the password for the Active Directory user account.
3. At the command line prompt, enter `id` to view the Unix account information.
4. After a successful log in, verify that the user obtained a Kerberos ticket by entering:

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

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

5. Enter `exit` to close the command shell.

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

Running reports

You can run various reports that capture key information about the Unix hosts you manage from the management 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 management console to profile hosts automatically. For more information, see [Configuring automatic profiling](#) on page 40.

2. From the management console, click the **Reporting** tab.
3. From the **Reports** view, expand the report group names to view the available reports, if necessary.
 - **Host Reports**
Unix host information gathered during the profiling process
 - **User Reports**
Local and Active Directory user information
 - **Group Reports**
Local and Active Directory group information
 - **Access & Privileges Reports**
User access information
 - **License Usage Reports**
Product licensing information.
4. Use one of the following methods to select a report:
 - Double-click a report name in the list (such as the **Unix Host Profiles** report).
 - Right-click a report name and select **Run report**.
 - Click the report icon  next to a report.

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

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

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

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

- On Windows:

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

- On Unix:

```
/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 *JVM memory tuning suggestions* in the *Management Console for Unix Administration Guide*.

Reports

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

NOTE: Report availability depends on several factors:

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

Host reports

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

Table 22: Host reports

Report	Description
Authentication	Provides a snapshot of the readiness of each host to join Active

Report	Description
Services Readiness	<p data-bbox="416 271 1398 331">Directory. This report is best used for planning and monitoring migration projects. The basic report includes the following information:</p> <ul data-bbox="467 353 1398 607" style="list-style-type: none"> <li data-bbox="467 353 788 383">• Total number of hosts <li data-bbox="467 405 1345 434">• Total number, percentage, and names of the hosts ready to join <li data-bbox="467 456 1345 517">• Total number, percentage, and names of the hosts ready to join with advisories <li data-bbox="467 539 1398 568">• Total number, percentage, and names of the hosts not ready to join <li data-bbox="467 591 1182 620">• Total number of hosts not checked for AD readiness <p data-bbox="416 636 1342 696">Use the following report parameters to define details to include in the report.</p> <ul data-bbox="467 725 919 943" style="list-style-type: none"> <li data-bbox="467 725 663 754">• Joined to AD <li data-bbox="467 777 722 806">• Ready to Join AD <li data-bbox="467 828 919 857">• Ready to Join AD with Warnings <li data-bbox="467 880 775 909">• Not Ready to Join AD <li data-bbox="467 931 855 960">• Not Checked for Readiness <p data-bbox="416 976 1374 1032">NOTE: This report is available when you are logged on as the supervisor or an Active Directory account in the Manage Hosts role.</p>
Privilege Manager Readiness	<p data-bbox="416 1061 1398 1122">Provides a snapshot of the readiness of each host to join a policy group. The basic report includes the following information:</p> <ul data-bbox="467 1151 1398 1317" style="list-style-type: none"> <li data-bbox="467 1151 788 1180">• Total number of hosts <li data-bbox="467 1202 1345 1232">• Total number, percentage, and names of the hosts ready to join <li data-bbox="467 1254 1398 1283">• Total number, percentage, and names of the hosts not ready to join <li data-bbox="467 1305 1139 1335">• Total number of hosts not checked for readiness <p data-bbox="416 1350 1342 1411">Use the following report parameters to define details to include in the report.</p> <ul data-bbox="467 1440 1062 1655" style="list-style-type: none"> <li data-bbox="467 1440 812 1469">• Joined to a policy group <li data-bbox="467 1491 868 1520">• Ready to join a policy group <li data-bbox="467 1543 1062 1572">• Ready to join a policy group with warnings <li data-bbox="467 1594 919 1624">• Not ready to join a policy group <li data-bbox="467 1646 844 1675">• Not checked for readiness <p data-bbox="416 1691 1294 1771">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.</p>
Unix	Lists all Unix computers in Active Directory in the requested scope.

Report	Description
Computers in AD	<p>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.</p> <p>NOTE: This report is available when you are logged on as an Active Directory account in the Manage Hosts role.</p>
Unix Host Profiles	<p>Summarizes information gathered during the profiling process of each managed host. This report includes the following information:</p> <ul style="list-style-type: none"> • Total number of hosts included in the report • Host Name, IP Address, OS, Hardware • Sudo version number <p>Use the following report parameters to define details to include for each host.</p> <ul style="list-style-type: none"> • Authentication Services Properties • Privilege Manager Properties • Local Users • Local Groups • Host SSH Keys • Installed One Identity Software <p>NOTE: This report is available when you are logged on as the supervisor or an Active Directory account in the Manage Hosts role.</p>

User reports

The following reports provide local and Active Directory user information.

Table 23: User reports

Report	Description
AD User Conflicts	<p>Returns all users with Unix User ID numbers (UID numbers) assigned to other Unix-enabled user accounts.</p> <p>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.</p> <p>NOTE: This report is available when you are logged on as an Active Directory account in the Manage Hosts role.</p>
Local Unix User Conflicts	<p>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</p>

Report	Description
	<p>information:</p> <ul style="list-style-type: none"> • Host Name, DNS Name, or IP Address where a conflict would occur • User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory, and Login Shell for each host where conflicts exist <p>Use the following report parameters to define the user name and UID number that would cause a conflict with existing local user accounts:</p> <ul style="list-style-type: none"> • User Name is • UID Number is <p>NOTE: This report is available when you are logged on as the supervisor or an Active Directory account in the <i>Manage Hosts</i> role.</p>
Local Unix Users	<p>Lists all users on all hosts or lists the hosts where a specific user account exists in <code>/etc/passwd</code>. This report includes the following information:</p> <ul style="list-style-type: none"> • Host Name, DNS Name, or IP Address where the user exists • User Name, UID Number, Primary GID Number, Comment (GECOS), Home Directory, and Login Shell for each host where the user exists <p>If you do not define a specific user, it includes all local users on each profiled host in the report.</p> <p>To locate a specific user, use the following report parameters:</p> <ul style="list-style-type: none"> • User Name contains • UID Number is • Primary GID Number is • Comment (GECOS) contains • Home Directory contains • Login Shell contains <p>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.</p> <p>NOTE: This report is available when you are logged on as the supervisor or an Active Directory account in the Manage Hosts role.</p>
Local Unix Users with AD Logon	<p>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:</p> <ul style="list-style-type: none"> • Host Name, DNS Name, or IP Address of hosts where users exist that are required to log on using their AD credentials • User Name, UID Number, Primary GID Number, and Comment

Report	Description
	<p>(GECOS) of local user account</p> <ul style="list-style-type: none"> The SAM account Name of the Active Directory account that the local user account must use to log on <p>NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.x agent.</p> <p>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.</p>
Master /etc/passwd List	<p>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:</p> <ul style="list-style-type: none"> Username Empty password UID GID GECOS Home directory path Account's shell <p>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.</p> <p>Indicate how you want to match user accounts by selecting the value parameters that you want to match:</p> <ul style="list-style-type: none"> Username UID GID GECOS Home Directory Shell <p>Optionally, you can include the host name for the accounts, as well:</p> <ul style="list-style-type: none"> Include the host name for accounts <p>This report is available when you are logged on as the supervisor or an Active Directory account in the Manage Hosts role.</p> <p>NOTE: If you select the Include the host name for accounts option,</p>

Report	Description
	<p>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 .csv file into the Unix Account Import Wizard, you must remove the Host column.</p> <p>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 Services Control 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.</p>
Unix-Enabled AD Users	<p>Lists all Active Directory users that have Unix user attributes.</p> <p>NOTE:</p> <ul style="list-style-type: none"> • A User object is considered to be 'Unix-enabled' if it has values for the UID Number, Primary GID Number, Home Directory, and Login Shell. • If Login Shell is /bin/false, the user is considered to be disabled for Unix or Linux logon. • Account Disabled indicates whether the Active Directory User account is enabled or disabled. <p>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.</p> <p>NOTE: This report is only available if you have configured the mangement console to recognize Active Directory objects (see <i>Configuring the Console to Recognize Unix Attributes in AD</i> in the online help), and you are logged on as an Active Directory account in the Manage Hosts role.</p>

Group reports

The following reports provide local and Active Directory group information.

Table 24: Group reports

Report	Description
AD Group	Lists all Active Directory groups with Unix Group ID (GID) numbers assigned to other Unix-enabled groups.

Report	Description
Conflicts	<p>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.</p> <p>NOTE: This report is available when you are logged on as an Active Directory account in the Manage Hosts role.</p>
Local Unix Groups	<p>Identifies the hosts where a specific group exists in /etc/group. This report includes the following information:</p> <ul style="list-style-type: none"> • Host Name, DNS Name, or IP Address where the group exists • Group Name, GID Number, and members for each host where the group exists <p>If you do not specify a group, it includes all local groups on each profiled host in the report.</p> <p>To locate a specific group, use the following report parameters:</p> <ul style="list-style-type: none"> • Group Name contains • GID Number is • Member contains • Include all group members in report <p>NOTE: The Member contains field accepts multiple entries separated by a comma. Spaces are taken literally in the search. For example, entering:</p> <ul style="list-style-type: none"> • adm, user searches for members whose name contains "adm" or "user" • adm,user searches for members whose name contains "adm" or "user" <p>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.</p> <p>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.</p> <p>NOTE: This report is available when you are logged on as the supervisor or an Active Directory account in the Manage Hosts role.</p>
Unix-Enabled AD Groups	<p>Lists all Active Directory groups that have Unix group attributes.</p> <p>NOTE: A Group object is considered 'Unix-enabled' if it has a value for the GID Number.</p> <p>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</p>

Report	Description
--------	-------------

base container to begin the search.

NOTE: This report is only available if you have configured the management console to recognize Active Directory objects (see *Configuring the Console to Recognize Unix Attributes in AD* in the online help), and you are logged on as an Active Directory account in the *Manage Hosts* role.

Access & Privileges reports

The following reports provide user access information.

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

Table 25: Access & Privileges reports

Report	Description
Access & Privileges by Host	<p>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:</p> <ul style="list-style-type: none">• Total number of users that can log on to the host• The users that can log on to the host• The commands users can run on the host• The runas aliases for which the user can run commands on the host• The commands the runas alias can run on the host <p>Browse to select a host.</p> <p>Optionally, select the Show detailed report option.</p> <p>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.</p>
Access & Privileges by User	<p>Identifies the users with logon access to hosts, the commands that user can run on each host, and the "runas aliases" information for that user. This report includes the following information:</p> <ul style="list-style-type: none">• Total number of hosts where the user can log on• The hosts where the user can log on• The commands the user can run on each host• The runas aliases for which the user can run commands on each host• The commands the runas alias can run on each host

Report	Description
	<p>Use the following report parameters to specify the user to include in the report:</p> <ul style="list-style-type: none"> • A local user (default) • An AD user <p>Browse to select a user.</p> <p>Optionally select the Show detailed report option.</p> <p>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.</p>
Commands Executed	<p>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:</p> <ul style="list-style-type: none"> • Command name • User who executed the command • Date and time the command was executed • Host where the command was executed <p>Use the following report parameters to define details in the report:</p> <ul style="list-style-type: none"> • Policy Group • Command • Host • Log status • Date <p>NOTE: You can use wildcards in the text string you enter in the Command box, such as * and ?.</p> <p>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.</p>
Console Access and Permissions	<p>Lists users who have access to the management console based on membership in a console role and the permissions assigned to that role. This report includes the following information:</p>

Report	Description
	<ul style="list-style-type: none"> • List of roles • List of permissions assigned to each role • List and number of members assigned to each role <p>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 management console requires that you authenticate to Active Directory.</p>
<p>Logon Policy for AD User</p>	<p>Identifies the hosts where Active Directory users have been granted logon permission. This report includes the following information for hosts joined to an Active Directory domain:</p> <ul style="list-style-type: none"> • Total number of hosts where the AD user has access • List of hosts where the AD user has access <p>Specify the Active Directory users to include in the report:</p> <ul style="list-style-type: none"> • All AD users (default) • Select AD user <p>Browse to search Active Directory to locate and select an Active Directory user.</p> <p>NOTE: The report may show both the Active Directory login name and local user names in the Login Name column for a selected AD user account because an Active Directory user account can have one or more local user accounts mapped to it.</p> <p>NOTE: Only hosts joined to an Active Directory domain with a Authentication Services 4.x agent are included in this report.</p> <p>NOTE: This report is available when you are logged on as an Active Directory account in the Manage Hosts role.</p>
<p>Logon Policy for Unix Host</p>	<p>Identifies the Active Directory users that have been explicitly granted logon permissions for one or more Unix computers. This report includes the following information for hosts joined to an Active Directory domain:</p> <ul style="list-style-type: none"> • Host Name, DNS Name, or IP Address of the host selected for the report • Users that have been granted permission to log on <p>Specify the managed hosts to include in the report:</p> <ul style="list-style-type: none"> • All profiled hosts (default) • Select host <p>Browse to locate and select a managed host that is joined to Active Directory.</p>

Report	Description
	<p>NOTE: This report only includes hosts joined to an Active Directory domain with a Authentication Services 4.x agent.</p> <p>NOTE: This report is available when you are logged on as an Active Directory account in the Manage Hosts role.</p>
Policy Changes	<p>Provides details of changes made to a policy for a Privilege Manager policy group. This report includes the following information:</p> <ul style="list-style-type: none"> • Name of the user that made changes to the policy • Version number for the changes • Time and date the changes were saved and actively used to enforce policy • Changes made to the policy based on version <p>Select a policy group.</p> <p>Select to:</p> <ul style="list-style-type: none"> • Show all changes to the policy • Show only changes for a specific pmpolicy file (not available for sudo-based policy) • Show changes to the policy for changes for one or more revisions <p>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.</p>

Product licenses usage report

The following report provides product licensing information.

Table 26: Product licenses usage reports

Report	Description
Product License Usage	<p>Provides a summary of all licensing information. This report includes the following information for hosts managed by the console:</p> <ul style="list-style-type: none"> • Product • Purchased licenses • Used licenses

Use Authentication Services PowerShell

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

You can perform the following tasks using PowerShell cmdlets:

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

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

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

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

To Unix-enable a user and user group

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

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

3. At the PowerShell prompt, enter the following:

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

NOTE: You created the UNIXusers group in a previous exercise. See [Adding an Active Directory group account](#) on page 74.

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 | Set-QasUnixUser -PrimaryGidNumber 1234567
```

The Unix properties of the user display:

```

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

```

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

```
Disable-QasUnixUser ADuser
```

NOTE: To clear all Unix attribute information, enter:

```
Clear-QasUnixUser ADuser
```

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

6. From the Control Center, under **Login to remote host**, enter:

- **Host name:** The Unix host name.
- **User name:** The Active Directory user name, **ADuser**.

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

A PuTTY window displays.

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

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

You will receive a message that says Access denied.

PowerShell cmdlets

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

Table 27: PowerShell cmdlets

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

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

cmdlet name	Description
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 macOS settings in Group Policy Objects
- Changes to product settings and configuration

Installing Change Auditor for Authentication Services

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

To install Change Auditor for Authentication Services

1. Insert the Authentication Services distribution media.
The Autorun **Home** page displays.
NOTE: If the Autorun **Home** page does not display, navigate to the root of the distribution media and double-click **autorun.exe**.
2. Click the **Setup** tab and select **Change Auditor for Authentication Services**.
The **Change Auditor for Authentication Services for Active Directory** web page opens.
3. Click **Download** on the left navigation panel.
4. Follow the online instructions to gain access to the **Trial Download** page.
5. From the **Trial Download: Change Auditor for Active Directory** page, click the **Installation Guide** link.

One Identity Defender

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

Installing Defender

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

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

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

To install Defender

1. Insert the Authentication Services distribution media.
The Autorun **Home** page displays.
NOTE: If the Autorun **Home** page does not display, navigate to the root of the distribution media and double-click **autorun.exe**.
2. From the **Home** page, click the **Setup** tab.
3. From the **Setup** tab, click **One Identity Defender**.
The **One Identity Defender** web page opens.
4. Click the **Download** on the left navigation panel.
5. Follow the online instructions to gain access to the **Trial Download** page.

6. From the **Trial Download: Defender** page, click the **Defender Documentation Archive** link.
7. Once you have installed One Identity Defender, see the *One Identity Defender Integration Guide* for detailed configuration instructions about integrating Authentication Services Defender with Authentication Services.

Troubleshooting

To help you troubleshoot, One Identity recommends the following resolutions to some of the common problems you might encounter as you deploy and use Authentication Services.

Getting help from technical support

If you are unable to determine the solution to a problem, contact Technical Support for help. For more information, see [About us](#).

Before you contact Support, please collect the following information:

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

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

This produces an output file in /tmp.

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

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

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

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

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

Disaster recovery

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

Long startup delays on Windows

You may experience long delays (over a minute) when starting the Authentication Services Windows installer or certain Windows management tools such as Control Center. All Authentication Services Windows binaries are Authenticode-signed so that you can be sure that the binaries are authentic and have not been tampered with. This problem occurs when the .NET runtime attempts to verify the Authenticode signature by checking against certificate revocation lists (CRLs) at `cr1.microsoft.com`. If this site cannot be reached, the .NET framework check will time out (up to 60 seconds). This timeout occurs every time a signed assembly is loaded which can lead to very long load times. You can fix this problem by allowing access to `cr1.microsoft.com`.

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

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

Pointer Record updates are rejected

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

Resolving DNS problems

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

be located using DNS:

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

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

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

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

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

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

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

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

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

Resolving preflight failures

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

Table 28: Install checks

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

Table 29: Join checks

Preflight option	Check	Resolution
--tld	Checks that the DNS Top Level Domain (TLD) is not '.local'.	Ensure that mDNS is disabled in /etc/nsswitch.conf or use a domain other than .local.
--hostname	Checks that the hostname of the system is not 'localhost'.	One Identity recommends that you have a unique hostname in order to maintain uniqueness of computer names in Active Directory. Another option is to ignore this check and use -n computer_name when joining. See the <i>vastool man page</i> for more information.
--name-service	Checks if the name service is configured to use DNS.	Ensure your host is configured to use DNS properly. Consult your platform documentation to determine the proper method to enable DNS for hostname resolution. See Resolving DNS problems on page 99 for solutions.
--host-resolve	Ensures that the host can resolve names using DNS.	Check your /etc/resolv.conf file to ensure that name server entries are correct and reachable. Make sure that UDP port 53 (DNS) is open. This check attempts to resolve the domain name and can fail if your DNS configuration is invalid. This check expects to find properly formatted IPv4 addresses. Invalid or unreachable name server entries will cause delays even though the check will pass if at least one valid name server is found. If you notice delays when running this check, make sure that your name

Preflight option	Check	Resolution
		server configuration does not reference invalid name servers. See Resolving DNS problems on page 99 for solutions.
--srv-records	Checks for a nameserver that has the appropriate DNS SRV records for Active Directory.	SRV records advertise various Active Directory services. Your configured name server must provide SRV records in order for Authentication Services to take advantage of automatic detection and fail over. Ensure that UDP port 53 (DNS) is open.
--dc	Detects a writable domain controller with UDP port 389 open.	<p>If a domain controller is passed on the preflight command line, <code>preflight</code> checks that UDP port 389 is open and that the domain controller is writable. In this case, you may be able to specify a different domain controller.</p> <p>If you do not pass in the name of a domain controller, this check attempts to locate a writable domain controller using DNS SRV records. Ensure that your DNS SRV records are up to date in the configured DNS server. Authentication Services can work with read-only domain controllers, but the computer object must have already been created with the proper settings in Active Directory.</p>
--site	Detects Active Directory site, if available.	This check warns you if Authentication Services was unable to locate an Active Directory site based on your computer's network address. A site configuration is not necessary, but Authentication Services performs better if site information is configured in Active Directory. To resolve this problem, configure a site in Active Directory.
--kerberos-password	Checks if TCP port 464 is open for Kerberos <code>kpasswd</code> .	Ensure that TCP port 464 (<code>kpasswd</code>) is open. This port must be open in order for Authentication Services to set the computer object's password.
--kerberos-traffic	Checks if UDP port 88 and TCP port 88 are open for Kerberos traffic.	These ports are the main Kerberos communication channels; they must be open for Authentication Services to authenticate to Active Directory. By default Authentication Services uses TCP, but may be configured to prefer UDP.
--ldap	Checks if TCP port 389 is	This port must be open for Authentication Services to communicate with domain controllers using LDAP. This

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

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

Table 30: Post-join checks

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

Time synchronization problems

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

```
vastool timesync
```

Unable to install or upgrade

The most common installation or upgrade failure is that the Unix host cannot read the Authentication Services application configuration in Active Directory. Ensure that you have followed the instructions in [Configure Active Directory for Authentication Services](#) on page 29 and that the configuration has been created successfully.

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

For more information, see [About Active Directory configuration](#) on page 31.

Unable to join the domain

If you are unable to join the domain, run the `preflight` utility to validate your environment. For more information, see [The Authentication Services pre-installation diagnostic tool](#) on page 50.

Then, verify the following:

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

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

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

For more information, see [About Active Directory configuration](#) on page 31.

- Properly configure DNS to return srv-records and avoid joining to a specific domain controller.

Unable to log in

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

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

```
vastool status
```

Correct any errors reported by the status command, then try logging in again.

3. Ensure the user exists locally and is allowed to log in. To check this, run the following command:

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

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

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

| **NOTE:**

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

vasypd has unsatisfied dependencies

If you receive the following error message while installing the Authentication Services vasynd Unix component, the rpcbind service may not be enabled.

```
svcadm: Instance "svc:/quest/vas/vasypd:default" has unsatisfied dependencies.  
Error 4 starting vasynd
```

To enable the rpcbind service

1. Check the dependencies of vasynd:

```
# svcs -d quest/vas/vasypd  
STATE      STIME    FMRI  
disabled   Sep_14   svc:/network/rpc/bind:default  
online     Sep_14   svc:/milestone/single-user:default  
online     Sep_14   svc:/system/filesystem/local:default
```

2. If rpcbind is disabled, run this command to enable it:

```
# /usr/sbin/svcadm enable -s /network/rpc/bind
```

3. Run the following command to start vasynd:

```
# /etc/init.d/vasypd start
```

Enterprise package deployment

This section details how to install, upgrade, and uninstall the Authentication Services agent on supported platforms in an enterprise environment using platform package management tools.

Installing the Authentication Services agent package

To install the Authentication Services agent package

1. Log in and open a root shell.
2. Mount the installation ISO and run the appropriate command.
See [Additional configuration information](#) that follows the table.

Table 31: Authentication Services: Agent commands

Platform	Command
Linux x86 - RPM	# rpm -Uhv /<mount>/client/linux-x86/vasclnt-<version>-<build>.i386.rpm
Linux x64 - RPM	# rpm -Uhv /<mount>/client/linux-x86_64/vasclnt-<version>-<build>.x86_64.rpm
Linux x86 - DEB	# dpkg -i /<mount>/client/linux-x86/vasclnt-<version>-<build>.i386.deb
Linux x64 - DEB	# dpkg -i /<mount>/client/linux-x86_64/vasclnt-<version>-<build>_amd64.deb
Linux s390	# rpm -Uhv /<mount>/client/linux-s390/vasclnt-<version>-<build>.s390.rpm
Linux s390x	# rpm -Uhv /<mount>/client/linux-s390x/vasclnt-<version>-

Platform	Command
	<build>.s390x.rpm
SLES 11, 12, and 15 PPC	# rpm -Uhv /<mount>/client/linux-glibc23-ppc64/vasclnt-glibc23-<version>-<build>.ppc64.rpm
Oracle Solaris 10 and 11 x64	# pkgadd -d /<mount>/client/solaris10-x64/vasclnt_SunOS_5.10_i386-<version>-<build>.pkg vasclnt
Oracle Solaris 10 and 11 SPARC	# pkgadd -d /<mount>/client/solaris10-sparc/vasclnt_SunOS_5.8_sparc-<version>-<build>.pkg vasclnt
HP-UX PA-RISC 11i v3 (B.11.31)	# swinstall -s /<mount>/client/hpux-pa-11v1/vasclnt_hpux-11.11-<version>-<build>.depot vasclnt
HP-UX IA64 11i v3 (B.11.31)	# swinstall -s /<mount>/client/hpux-ia64/vasclnt_ia64-<version>-<build>.depot vasclnt
AIX 7.1 and 7.2	# installp -acXd /<mount>/client/aix-71/vasclnt.AIX_5.3.<version>-<build>.bff all
Mac OS X	/usr/sbin/installer -pkg '/<mount>/VAS.mpkg/Contents/Packages/vasclnt.pkg' -target /
FreeBSD 10 and 11	pkg /<mount>/client/freebsd-x86_64/vasclnt-<build>.txz
Amazon Linux AMI	# rpm -Uhv /<mount>/client/linux-x86_64/vasclnt-<build>.x86_64.rpm

Additional configuration information:

- To enable Authentication Services authentication for all services, you must restart all services that require Authentication Services authentication or restart the system.
- Linux - RPM:** The x86_64 Authentication Services rpm contains 64-bit and 32-bit libraries, and has an RPM dependency on both the 32-bit libpam library and the 64-bit libpam library. If the 64-bit Linux operating system on which you are installing Authentication Services does not have any 32-bit supporting libraries installed, use the -- nodeps RPM flag to force the installation and avoid error messages about missing dependencies.
- Oracle Solaris:** For information on Oracle Solaris 10 Zones support and installation, see [Oracle Solaris 10 zones/containers support](#) on page 112.

In certain situations, pkgadd requests additional information. Respond appropriately for your system configuration. Initialization scripts that are part of the vasclnt package run during installation to help configure the system.

To install the Authentication Services vasydpd Unix component on Oracle Solaris 10, you must have the rpcbind service enabled on the host. See [vasydpd has unsatisfied dependencies](#) on page 106 for more information.

- **HP-UX:** Authentication Services requires that the Unixhost system clock be synchronized with the Active Directory server's system clock. By default, HP-UX uses xntpd for time services. To properly synchronize the system clocks either configure xntpd to sync with a Domain Controller, or disable xntpd to allow Authentication Services to synchronize the system time. Consult the xntpd documentation for information on disabling xntpd and configuring xntpd.

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

- **macOS:** To install from the command line, you must first mount the Authentication Services DMG image file. On macOS enter:

```
hdiutil attach <media>/client/macos-106/VAS-<version>.dmg
```

Authentication Services agent upgrade commands

To upgrade the Authentication Services agent package

1. Log in and open a root shell.
2. Mount the installation ISO and run the appropriate command.
See [Additional configuration information](#) that follows the table.

Table 32: Authentication Services: Agent commands

Platform	Command
Linux x86 - RPM	# rpm -Uhv /<mount>/client/linux-x86/vasclnt-<version>-<build>.i386.rpm
Linux x64 - RPM	# rpm -Uhv /<mount>/client/linux-x86_64/vasclnt-<version>-<build>.x86_64.rpm
Linux x86 - DEB	# dpkg -i /<mount>/client/linux-x86/vasclnt-<version>-<build>.i386.deb
Linux x64 - DEB	# dpkg -i /<mount>/client/linux-x86_64/vasclnt-<version>-<build>_amd64.deb
Linux s390	# rpm -Uhv /<mount>/client/linux-s390/vasclnt-<version>-<build>.s390.rpm
Linux s390x	# rpm -Uhv /<mount>/client/linux-s390x/vasclnt-<version>-<build>.s390x.rpm
SLES 11, 12, and	# rpm -Uhv /<mount>/client/linux-glibc23-ppc64/vasclnt-

Platform	Command
15 PPC	<code>glibc23-<version>-<build>.ppc64.rpm</code>
Oracle Solaris 10 and 11 x64	<code># pkgadd -d /<mount>/client/solaris10-x64/vasclnt_SunOS_5.10_i386-<version>-<build>.pkg vasclnt</code>
Oracle Solaris 10 and 11 SPARC	<code># pkgadd -d /<mount>/client/solaris10-sparc/vasclnt_SunOS_5.8_sparc-<version>-<build>.pkg vasclnt</code>
HP-UX PA-RISC 11i v3 (B.11.31)	<code># swinstall -s /<mount>/client/hpux-pa-11v1/vasclnt_hpux-11.11-<version>-<build>.depot vasclnt</code>
HP-UX IA64 11i v3 (B.11.31)	<code># swinstall -s /<mount>/client/hpux-ia64/vasclnt_ia64-<version>-<build>.depot vasclnt</code>
AIX 7.1 and 7.2	<code># installp -acXd /<mount>/client/aix-71/vasclnt.AIX_5.3.<version>-<build>.bff all</code>
Mac OS X	<code>/usr/sbin/installer -pkg '/<mount>/VAS.mpkg/Contents/Packages/vasclnt.pkg' -target /</code>
FreeBSD 10 and 11	<code>pkg /<mount>/client/freebsd-x86_64/vasclnt-<build>.txz</code>
Amazon Linux AMI	<code># rpm -Uhv /<mount>/client/linux-x86_64/vasclnt-<build>.x86_64.rpm</code>

Additional configuration information

NOTE: During the upgrade, `vasd` reloads and updates its user and group cache. To restart the Authentication Services caching service, see [Restarting Authentication Services services](#) on page 111.

NOTE: If you are using the licensed version of the Authentication Services agent earlier than 3.0, see [Licensing Authentication Services](#) on page 55 for licensing instructions.

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

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

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

Restarting Authentication Services services

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

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

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

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

- c. To restart Authentication Services on AIX, enter:

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

NOTE: Due to library changes between the Authentication Services 4.1 and 4.2, the system may need to be rebooted before all processes load the new libraries.

Uninstalling the Authentication Services agent packages

To uninstall the Authentication Services agent packages

1. Log in and open a root shell.
2. Run the following commands to remove the packages.
See [Additional configuration information](#) that follows the table.

Table 33: Authentication Services: Agent uninstall commands

Package	Command
RPM	# rpm -e vasclnt
DEB	# dpkg -r vasclnt
Oracle Solaris	# pkgrm vasclnt
HP-UX	# swremove vasclnt
AIX	# installp -u vasclnt
macOS	/<mount>/Uninstall.app/Contents/MacOS/Uninstall' --console --force vasclnt

Package	Command
FreeBDS	pkg delete <package name>

Additional configuration information

- **Linux:** The `rpm -e vasclnt` and the `dpkg -r vaslcnt` commands run scripts that halt the daemon, unconfigure Authentication Services, flush, and delete the Authentication Services cache before finally removing the files.
- **HP-UX:** The `swremove vasclnt` command does not clean up the empty directories that the `vasclnt` package used. In order to clean these up, manually remove the `/opt/quest` directory after you uninstall.

Oracle Solaris 10 zones/containers support

Zones (or containers) were introduced in Oracle Solaris 10. Zones is a partitioning technology used to virtualize operating system services and provide an isolated and secure environment for running applications. There are two types of non-global zone root filesystem models:

- sparse root
- whole root

The sparse root zone model optimizes the sharing of objects while the whole root zone model provides the maximum configurability. Additional information on Oracle Solaris 10 and Zones can be found at www.sun.com.

Authentication Services and Oracle Solaris 10 Zones installation guidelines

To install Authentication Services in a Oracle Solaris 10 Zones configuration

- In Oracle Solaris 10 Zones, only the global zone is permitted to do time synchronization. Therefore, if you want to run Authentication Services in any Oracle Solaris Zone configuration, you must `timesync` the Global Zone with Active Directory. Time synchronization is a requirement of the Kerberos protocol and since Authentication Services is built on Kerberos, Authentication Services also has this requirement.
- The same version of Authentication Services should be installed in any combination of global, whole root, and sparse root zone configurations.

- To disable time synchronization for Authentication Services on the sparse zone, run the below command:

```
vastool configure vas vasd timesync-interval 0
```

- The following symlinks must exist in the global zone in order for the sparse zones to work correctly:

- /usr/lib/security/pam_vas3.so | /opt/quest/usr/lib/security/pam_vas3.so
- /usr/lib/security/sparcv9/pam_vas3.so |
/opt/quest/usr/lib/security/sparcv9/pam_vas3.so

If /usr is shared, you need the following symlinks in the global zone pointing to counterpart files in /opt/quest/lib:

- /usr/lib/nss_vas4.so.1 | /opt/quest/lib/nss/nss_vas4.so.1
- /usr/lib/security/pam_vas3.so | /opt/quest/usr/lib/security/pam_vas3.so

In such a scenario, you do not need Authentication Services joined to a domain in the global zone in order for sparse zones to work, but the symlinks must exist.

Each zone must have its own unique copy of /etc and /var because Authentication Services stores zone-specific information in those locations. Sharing /etc and /var with the global zone is not a supported configuration.

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 and other inquiries, such as licensing, support, and renewals, visit <https://www.oneidentity.com/company/contact-us.aspx>.

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

A

- Access & Privileges by Host report 87
- Access & Privileges by User report 87
- Access & Privileges Reports 87
- Active Directory configuration
 - changing configuration settings 12
 - determines schema mappings 31
 - moving the configuration data 31
 - purpose defined 31
 - updating 31
 - validates license information 31
- Active Directory group account
 - creating 74
- Active Directory schema
 - how uses 70
- Active Directory user account
 - creating 74
- ActiveRoles Server option
 - not available if ActiveRoles Server agent is not installed 29
- AD Group Conflicts report 85
- AD User Conflicts report 82
- AD user identity formats 37
- AD users and groups
 - managing 75
- Add Hosts
 - procedure 38
- Add Hosts dialog
 - add hosts to management console 38
 - profile hosts 39
- agent packages
 - uninstalling 111
 - upgrading 109
- agent upgrade commands 109
- All Hosts dialog
 - Join to Active Directory 45
- All Hosts view
 - install software 43
- application configuration
 - creating from command line 57
 - overriding requirement 32
 - running Authentication Services without 33
- application integration
 - installing Change Auditor 95
 - installing Defender 96
- associate an AD user account with a local Unix user 75
- Authentication Services
 - configure management console 34-36
- Authentication Services Readiness report 80
- automatic profiling
 - disable 40
 - enable 40
- automatic status checking
 - disable 47
 - enable 47

B

- Best Practice:
 - add Unix identity attributes to global catalog 71

- do not install or run Windows components on AD domain controllers 11
- index attributes in Active Directory 71
- install only one management console per environment 62
- license management 55
- use generated UUIDs and GIDs 67
- use schema designed for storing Unix data in AD 69
- Version 3 Compatibility Mode 33

C

- caching of Unix host credentials 39
- change Active Directory configuration settings 31
- Change Auditor integration
 - installing 95
- Check for AD Readiness 45
- check QAS Agent Status 47
- check QAS status automatically 47
- Commands Executed report 87
- components
 - .msi files 26
 - installing windows components with msiexec.exe 26
- configure
 - user service account 40, 47
- configure for Active Directory 34
- configure for Authentication Services 35-36
- Console Access and Permissions report 87
- Control Center
 - adding license 66
 - described 62

- must be logged in as domain user 62
- credentials
 - accepted user name formats 37
- customize the schema mapping 71

D

- debug logging
 - enabling 69
- Defender
 - installing 96
- disable automatic profiling 40
- Dynamic DNS Update Tool 60

E

- Edit GPO 64
- elevated credentials required
 - automatic profiling 40
 - automatic QAS status 47
 - install Authentication Services software 43
 - join host to AD 45
- enable debug logging 69
- enable local user for AD authentication 75

F

- Filter Options 64

G

- Generating 64
- global settings modifications 62
- Global Unix Options
 - Unique IDs 67
 - Unix group defaults 67

- Unix user defaults 67
 - where to set 67
- group
 - add to console 72
- Group Policy
 - editing GPO 64
 - filtering list of GPOs 64
 - launching GPMC 65
 - showing templates for GPO 65
- Group Reports 85

H

- health status of QAS agents
 - checking 46
- host
 - check status 47
 - join to Active Directory 45
- Host Notifications 47
- Host Reports 80
- hosts
 - add to management console 38
 - install software 43
 - profile 39-40

I

- Import Public Key
 - using 39
- Install Software
 - procedure 43
- install.sh
 - about 52
 - interactive mode 60
- installation script
 - using 53

- Installation Script Options 53
- installing
 - Linux agent 107
 - MSI properties 26
 - Unix agent 50

J

- join domain in Version 3 Compatibility Mode 33
- Join Host procedure 45
- joining domain
 - determining if joined 58
- joining the AD domain 58-59

K

- known_hosts file
 - importing 38

L

- Launch GPMC 65
- LDAP attributes
 - mapped to Unix attributes 69
- license
 - adding using Control Center 66
 - Any VAS 3.x or higher license is valid for 4.x. 10
 - installing 10
 - installing from command line 56
 - managing licenses 55
 - updating 36
 - updating in the console 10
- Limitations
 - Microsoft does not support (GPMC) on 64-bit platforms of Windows 11

- Linux agent
 - installing 107
- local account administration 72-73
- Local Unix Groups report 85
- Local Unix User Conflicts report 82
- Local Unix Users report 82
- Local Unix Users with AD Logon report 82
- logging
 - enabling 69
 - enabling debug logging in Windows 69
 - setting options 69
- login credentials
 - accepted formats 37
- Login with AD password 76, 78
- Logon Policy for AD User report 87
- Logon Policy for Unix Host report 87

M

- manage local users and groups 75
- management console
 - add hosts 38
 - requirements 20
- mapping users 75
- Master /etc/passwd List report 82
- migrating Unix account info to AD 80

O

- Optimize Schema
 - requires AD administrator rights 71

P

- patch level requirements 13

- performance and scalability 71
- Permissions
 - required 12, 15
- Policy Changes report 87
- PosixAccount auxiliary class schema extension 70
- post-install setup 34
- PowerShell cmdlets 93
- PowerShell modules 91
- Preferences
 - configuring settings 66
 - Custom Unix Attributes 69
 - Global Unix Options 67
 - Licensing 66
 - Logging Options 69
- preflight Diagnostic Tool
 - About 50
- preflight utility
 - running 52
- Privilege Manager Readiness report 80
- Profile Host
 - procedure 39
- profile hosts automatically 40
- PTR updates are rejected 99

Q

- QAS Agent Status commands are not enabled 46
- questusr
 - about 40, 47

R

- reload configuration settings 111
- report
 - Access & Privileges by Host 87

- Access & Privileges by User 87
 - AD Group Conflicts 85
 - AD User Conflicts 82
 - Authentication Services Readiness 80
 - Commands Executed 87
 - Console Access and Permissions 87
 - descriptions 80
 - Local Unix Groups 85
 - Local Unix User Conflicts 82
 - Local Unix Users 82
 - Local Unix Users with AD Logon 82
 - Logon Policy for AD User 87
 - Logon Policy for Unix Host 87
 - Master /etc/passwd List 82
 - Policy Changes 87
 - Privilege Manager Readiness 80
 - Product Licenses Usage 90
 - report parameters 80
 - run 78
 - Unix-Enabled AD Groups 85
 - Unix-Enabled AD Users 82
 - Unix Computers in AD 80
 - Unix Host Profiles 80
 - required AD rights 62
 - required rights 31
 - Requirements:
 - encryption types 20
 - network ports 21
 - Permissions 15
 - Windows Management Tools 11
 - Windows Permissions 12
 - restart services 111
 - run reports 78
- S**
- saving credentials on server 39
 - saving host credentials on server 45
 - schema
 - configuration 69
 - Custom Unix attributes 69
 - extensions 69
 - LDAP attributes 69
 - templates 69
 - Unix attributes 69
 - schema configuration
 - defined 70
 - schema extension
 - PosixAccount auxiliary class 70
 - schema mappings
 - customizing
 - index and replicate GUI and UID attributes to global catalog 71
 - schemaless configuration 57
 - set global value 67
 - Set supervisor Password dialog 36
 - Setup Management Console for Unix dialog 34
 - Show Files 65
 - standard Active Directory schema extensions 70
 - supervisor account
 - described 36
- T**
- troubleshooting 98
 - using logs 69

Troubleshooting:

- cannot configure console for AD during initial install 34
- changes made to NSS libraries 111
- determine if joined to AD 58
- Diagnostic Tool 50
- feature not available 45
- Getting Help from Support 98
- join to AD is not available 45
- Long Startup Delays on Windows 99
- Profile Automatically option is not available 40
- PTR updates are rejected 60
- QAS status checking not working 46
- rejected PTR updates 99
- Resolving DNS Problems 99
- Resolving Preflight Failures 100
- Time Synchronization Problems 104
- Unable to Install or Upgrade 104
- Unable to Join the Domain 104
- Unable to Log In 105
- vasypd has unsatisfied dependencies 106

U

- uninstalling agent packages 111
- Unix-enable an Active Directory group 77
- Unix-enable an Active Directory user 77
- Unix-Enabled AD Groups report 85
- Unix-Enabled AD Users report 82
- Unix Account Import Wizard
 - accessing 80
- Unix agent
 - about installing 50
 - packages 107

requirements 13

- Unix attributes
 - changing default attributes 75
- Unix Computers in AD report 80
- Unix Group ID (GID)
 - set global value 67
- Unix Host Profiles report 80
- Unix identity management tasks
 - performing from Control Center 33
- Unix User ID (UID)
 - set global value 67
- upgrading VAS 3.5 to QAS 4.x 52
- User Reports 82
- user service account 40, 47
 - configure 40, 47
- users
 - add to console 73

V

- vas_status.sh
 - customizing 46
- vasd
 - restart 111
- vasjoin Script
 - using 59
- vasjoin.sh
 - using 58
- vastool join
 - using 58
- Version 3 Compatibility Mode 33