



TAMOS Agent Administration Guide



TAMOS Agent Administration Guide

Note

Before using this information and the product it supports, read the information in Appendix C, “Notices,” on page 91.

This edition applies to version 6, release 0, modification 0 of IBM Tivoli Access Manager for Operating Systems (product number 5698-PDO) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2008.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	v
Intended audience	v
Publications	v
IBM Tivoli Access Manager for Operating Systems library	v
IBM Tivoli Access Manager for e-business library	vi
Related products and publications.	vii
Accessing terminology online	ix
Accessing publications online	ix
Ordering publications	ix
Accessibility	x
Tivoli technical training.	x
Support information.	x
Conventions used in this publication	x
Typeface conventions	x

Chapter 1. Introduction	1
Introducing the TAMOS Agent	1
New features in TAMOS Agent	2
Supported platforms.	3
Environment prerequisites	4
Package contents	4

Chapter 2. Installation and Configuration	5
Before you start	5
System Considerations	5
Required information	5
Considerations when deploying in a Tivoli Access Manager version 6.1 environment	6
Considerations when using Solaris	6
Considerations when using AIX	8
Installation	9
Installing the prerequisite software	9
Extracting the install package	10
Installing the agent	10
Configuration.	11
Configuring the Tivoli Access Manager Runtime	11
Configuring the agent	12
Starting the agent	12
Stopping the agent	13
Unconfiguring the agent	13
Uninstalling the agent	14

Chapter 3. Defining Policy	15
Policy administration	15
Protected Object name structure and access controls	16
Protected system resources	16

Sudo policy	16
Login policy	21
Password management policy	30
File policy	35
Audit policy	48

Chapter 4. Comparisons with Tivoli Access Manager for Operating Systems 6.0	57
Policy support	57
Runtime	60
Daemon processes	60
PAM Module	61
Utilities.	61
Tivoli Access Manager for Operating Systems 6.0 utilities not included with TAMOS Agent	63
Tivoli Access Manager for Operating Systems 6.0 Kernel extensions and related utilities not included	64
TAMOS Agent Auditing	64
Operating System Files modified	65
Tivoli Access Manager for Operating Systems 6.0 Management Tasks component not included	67

Chapter 5. Troubleshooting	69
Known issues and limitations	69
Pluggable Authentication Module Parameters	70

Appendix A. Command reference	71
pdoscfg.	72
pdosucfg	82

Appendix B. Support information	85
Searching knowledge bases	85
Searching information centers	85
Searching the Internet	85
Obtaining fixes	85
Registering with IBM Software Support	86
Receiving weekly software updates	86
Contacting IBM Software Support	87
Determining the business impact	87
Describing problems and gathering information	88
Submitting problems	88

Appendix C. Notices	91
Trademarks	93

Preface

This publication describes the TAMOS Agent, which protects system resources by enforcing an authorization policy defined in terms of Tivoli® Access Manager access controls.

This publication introduces the agent, describes how to install and configure it with your Tivoli Access Manager for Operating Systems deployment, and provides other helpful information including auditing, policy definition and troubleshooting steps.

Intended audience

This guide is for system administrators responsible for the administration of IBM Tivoli Access Manager for Operating Systems. Readers should be familiar with the following:

- Microsoft® Windows® and UNIX® operating systems
- Security management
- Internet protocols, including HTTP, HTTPS, and TCP/IP
- Lightweight Directory Access Protocol (LDAP) and directory services
- Authentication and authorization
- The Tivoli Access Manager for Operating Systems security model

If you are enabling Secure Sockets Layer (SSL) communication, you also should be familiar with SSL protocol, key exchange (public and private), digital signatures, cryptographic algorithms, and certificate authorities.

Publications

This section lists publications in the IBM® Tivoli Access Manager library and related documents. The section also describes how to access Tivoli publications online, and how to order Tivoli publications.

IBM Tivoli Access Manager for Operating Systems library

- *IBM Tivoli Access Manager for Operating Systems Installation Guide*
Explains how to install, configure, and upgrade Tivoli Access Manager for Operating Systems software.
- *IBM Tivoli Access Manager for Operating Systems Administration Guide*
Describes the concepts and procedures for using Tivoli Access Manager for Operating Systems services.
- *IBM Tivoli Access Manager for Operating Systems Problem Determination Guide*
Describes how to troubleshoot and fix problems with your Tivoli Access Manager for Operating Systems installation.
- *IBM Tivoli Access Manager for Operating Systems Release Notes®*
Provides an overview of the changes, information regarding the product documentation, and workarounds for any known problems.

IBM Tivoli Access Manager for e-business library

Review the descriptions of the Tivoli Access Manager for Operating Systems library, the prerequisite publications, and the related publications to determine which publications you might find helpful. After you determine the publications you need, refer to the instructions for accessing publications online.

Additional information about the Tivoli Access Manager for e-business product itself can be found at the following Web address:

<http://www.ibm.com/software/tivoli/products/access-mgr-e-bus>

The Tivoli Access Manager for Operating Systems library is organized into the following categories:

- “Release information”
- “Installation and upgrade documentation”
- “Administration documentation”
- “Reference documentation” on page vii
- “Problem determination documentation” on page vii
- “Performance tuning documentation” on page vii

Release information

- *IBM Tivoli Access Manager for e-business: Release Notes*, GC23-6501-00
Provides information about installing and getting started, system requirements, known installation and configuration problems, and problem workarounds.

Installation and upgrade documentation

- *IBM Tivoli Access Manager for e-business: Installation Guide*, GC23-6502-00
Explains how to install and configure Tivoli Access Manager for e-business.
- *IBM Tivoli Access Manager for e-business: Upgrade Guide*, SC23-6503-00
Explains how to upgrade to Tivoli Access Manager for e-business version 6.0.
- *IBM Tivoli Access Manager for e-business: Quick Start Guide*, GI11-8174-00
Provides a high-level overview of a Tivoli Access Manager for e-business version 6.0 installation.

Administration documentation

- *IBM Tivoli Access Manager for e-business: Administration Guide*, SC23-6504-00
Describes the concepts and procedures for using Tivoli Access Manager for Operating Systems. Provides instructions for performing tasks from the Web Portal Manager interface and by using the **pdadmin** utility.
- *IBM Global Security Kit: Secure Sockets Layer Introduction and iKeyman User's Guide*, SC23-6510-00
Provides information for network or system security administrators who plan to enable SSL communication in their Tivoli Access Manager for Operating Systems environment.
- *IBM Tivoli Access Manager for e-business: Auditing Guide*, SC23-6511-00
Provides information about configuring and managing audit events using the native Tivoli Access Manager for Operating Systems approach and the Common Auditing and Reporting Service. Information about installing and configuring the Common Auditing and Reporting Service that can be used for generating and viewing operational reports is also provided.

Reference documentation

- *IBM Tivoli Access Manager for e-business: Command Reference*, SC23-6512-00
Provides reference information about the commands, utilities, and scripts that are provided with Tivoli Access Manager for Operating Systems.
- *IBM Tivoli Access Manager for e-business: Administration C API Developer Reference*, SC23-6513-00
Provides reference information about using the C language implementation of the administration API to enable an application to perform Tivoli Access Manager for Operating Systems administration tasks.
- *IBM Tivoli Access Manager for e-business: Administration Java Classes Developer Reference*, SC23-6514-00
Provides reference information about using the Java™ language implementation of the administration API to enable an application to perform Tivoli Access Manager for Operating Systems administration tasks.
- *IBM Tivoli Access Manager for e-business: Authorization C API Developer Reference*, SC23-6515-00
Provides reference information about using the C language implementation of the authorization API to enable an application to use Tivoli Access Manager for Operating Systems security.
- *IBM Tivoli Access Manager for e-business: Authorization Java Classes Developer Reference*, SC23-6516-00
Provides reference information about using the Java language implementation of the authorization API to enable an application to use Tivoli Access Manager for Operating Systems security.
- *IBM Tivoli Access Manager for e-business: Web Security Developer Reference*, SC23-6517-00
Provides programming and reference information for developing authentication modules.

Problem determination documentation

- *IBM Tivoli Access Manager for e-business: Problem Determination Guide*, GI11-8156-00
Provides problem determination information for Tivoli Access Manager for Operating Systems.
- *IBM Tivoli Access Manager for e-business: Error Message Reference*, GI11-8157-00
Provides explanations and recommended actions for the messages and return code that are generated by Tivoli Access Manager for Operating Systems.

Performance tuning documentation

- *IBM Tivoli Access Manager for e-business: Performance Tuning Guide*, SC23-6518-00
Provides performance tuning information for an environment consisting of Tivoli Access Manager for Operating Systems with the IBM Tivoli Directory Server as the user registry.

Related products and publications

This section lists the IBM products that are related to and included with a Tivoli Access Manager for Operating Systems solution.

IBM Global Security Kit

Tivoli Access Manager for Operating Systems provides data encryption through the use of the Global Security Kit (GSKit) version 7.0. GSKit is included on the *IBM Tivoli Access Manager Base* CD for your particular platform, as well as on the *IBM*

Tivoli Access Manager Web Security CDs, the IBM Tivoli Access Manager Shared Session Management CDs, and the IBM Tivoli Access Manager Directory Server CDs.

The GSKit package provides the iKeyman key management utility, **gsk7ikm**, which is used to create key databases, public-private key pairs, and certificate requests. The *IBM Global Security Kit: Secure Sockets Layer Introduction and iKeyman User's Guide* is available on the Tivoli Information Center Web site in the same section as the Tivoli Access Manager for Operating Systems product documentation.

IBM Tivoli Directory Server

IBM Tivoli Directory Server version 6.1 is included on the *IBM Tivoli Access Manager Directory Server* set of CDs for the desired operating system.

Additional information about Tivoli Directory Server can be found at the following Web address:

<http://www.ibm.com/software/tivoli/products/directory-server/>

IBM Tivoli Directory Integrator

IBM Tivoli Directory Integrator version 6.1.1 is included on the IBM Tivoli Directory Integrator CD for the desired operating system.

Additional information about IBM Tivoli Directory Integrator can be found at the following Web address:

<http://www-306.ibm.com/software/tivoli/products/directory-integrator/>

IBM DB2 Universal Database

IBM DB2[®] Universal Database[™] Enterprise Server Edition version 9.1 is provided on the *IBM Tivoli Access Manager Directory Server* set of CDs and is installed with the Tivoli Directory Server software. DB2[®] is required when using Tivoli Directory Server or z/OS[®] LDAP servers as the user registry for Tivoli Access Manager for Operating Systems. For z/OS LDAP servers, you must separately purchase DB2.

Additional information about DB2 can be found at the following Web address:

<http://www.ibm.com/software/data/db2>

IBM WebSphere Application Server

WebSphere[®] Application Server version 6.1 is included on the *IBM Tivoli Access Manager WebSphere Application Server* set of CDs for the desired operating system. WebSphere Application Server enables the support of the Web Portal Manager interface, which is used to administer Tivoli Access Manager for Operating Systems; the Web Administration Tool, which is used to administer Tivoli Directory Server; the Common Auditing and Reporting Service, which is used to process and report on audit events; the session management server, which is used to managed shared session in a Web security server environment and the Attribute Retrieval Service.

Additional information about WebSphere Application Server can be found at the following Web address:

<http://www.ibm.com/software/webservers/appserv/infocenter.html>

Accessing terminology online

The *Tivoli Software Glossary* includes definitions for many of the technical terms related to Tivoli software. The *Tivoli Software Glossary* is available at the following Tivoli software library Web site:

<http://publib.boulder.ibm.com/tividd/glossary/tivoliglossarymst.htm>

The IBM Terminology Web site consolidates the terminology from IBM product libraries in one convenient location. You can access the Terminology Web site at the following Web address:

<http://www.ibm.com/software/globalization/terminology>

Accessing publications online

The Tivoli Software Library provides a variety of Tivoli publications such as white papers, data sheets, demonstrations, Redbooks®, and announcement letters. The publications for this product and many other Tivoli products are available online in Portable Document Format (PDF) or Hypertext Markup Language (HTML) format, or both in the Tivoli software library at the following Web address:

<http://publib.boulder.ibm.com/tividd/td/tdprodlist.html>

To locate product publications in the library, click the first letter of the product name or scroll until you find the product name. Then click the name of the product. Product publications include release notes, installation guides, user's guides, administrator's guides, and developer's references.

Note: To ensure proper printing of PDF publications, select the **Fit to page** check box in the Adobe® Acrobat Print window (which is available when you click **File → Print**).

Ordering publications

You can order many Tivoli publications online at <http://www.elink.ibm.link.ibm.com/publications/servlet/pbi.wss>.

You can also order by telephone by calling one of these numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968

In other countries, contact your software account representative to order Tivoli publications. To locate the telephone number of your local representative, perform the following steps:

1. Go to <http://www.elink.ibm.link.ibm.com/publications/servlet/pbi.wss>.
2. Select your country from the list and click **Go**.
3. Click **About this site** in the main panel to see an information page that includes the telephone number of your local representative.

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

Tivoli technical training

For Tivoli technical training information, refer to the following IBM Tivoli Education Web site at <http://www.ibm.com/software/tivoli/education>.

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:

Online

Go to the IBM Software Support site at <http://www.ibm.com/software/support> and follow the instructions.

IBM Support Assistant

The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, go to <http://www.ibm.com/software/support/isa>.

Problem Determination Guide

For more information about resolving problems, see the *IBM Tivoli Access Manager for e-business: Problem Determination Guide*.

Conventions used in this publication

This publication uses several conventions for special terms and actions, operating system-dependent commands and paths, and margin graphics.

Typeface conventions

This publication uses the following typeface conventions:

Bold

- Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as **Tip:**, and **Operating system considerations:**)
- Keywords and parameters in text

Italic

- Citations (examples: titles of publications, diskettes, and CDs)
- Words defined in text (example: a nonswitched line is called a *point-to-point line*)

- Emphasis of words and letters (words as words example: "Use the word *that* to introduce a restrictive clause."; letters as letters example: "The LUN address must start with the letter *L*.")
- New terms in text (except in a definition list): a *view* is a frame in a workspace that contains data.
- Variables and values you must provide: ... where *myname* represents....

Monospace

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

Chapter 1. Introduction

This chapter contains the following sections:

- “Introducing the TAMOS Agent”
- “Supported platforms” on page 3
- “Environment prerequisites” on page 4
- “Package contents” on page 4

Introducing the TAMOS Agent

The TAMOS Agent extends the support of the Tivoli Access Manager for Operating Systems product into environments using Solaris 10 zones and AIX® 6.1 Workload Partitions (WPARs). The existing Tivoli Access Manager for Operating Systems product is not supported on systems using these types of virtualized operating system environments.

Solaris zones and AIX WPARs are software created, virtualized operating system environments, within a single instance of the native operating system. The TAMOS Agent provides support for these environments with a purely user-level endpoint agent, which translates policy that is expressed as Tivoli Access Manager for Operating Systems policy into native operating system security settings. This is very different from the existing Tivoli Access Manager for Operating product, which utilizes system kernel extensions to actively protect system resources.

Not all Tivoli Access Manager for Operating Systems policy is able to be translated to system settings. Access controls related to the following items are mapped to native operating system security and audit settings:

- file system resources (File policy),
- login services (Login activity policy),
- password management (Password management policy), and
- resource-level and user-level audit policy (POPs attached to file resources, **AuditAuth** and **AuditTrace** policy).

The TAMOS Agent continues to make authorization decisions for **Sudo**, **login location**, and **login holiday** policy. Other Tivoli Access Manager for Operating Systems authorization policies are not supported by the TAMOS Agent, such as policy related to network incoming and outgoing connections, or change of user and group identity (surrogate) operations.

Just like the existing Tivoli Access Manager for Operating Systems version 6.0 product, the TAMOS Agent functions in a Tivoli Access Manager secure domain environment that provides a backbone for centrally defining and managing security policy.

The TAMOS Agent also includes a new Native Policy Manager daemon to support mapping policy to system settings, and a new Pluggable Authentication Module (PAM) to support **login location** and **login holiday** policy. These are detailed in the following sections.

For a summary of Tivoli Access Manager for Operating Systems policy that is supported by the TAMOS Agent, see Table 35 on page 58.

For more information about differences with Tivoli Access Manager for Operating Systems version 6.0, see Chapter 4, “Comparisons with Tivoli Access Manager for Operating Systems 6.0,” on page 57.

For details on defining Tivoli Access Manager for Operating Systems policy for a TAMOS Agent endpoint, and how the policy is mapped by the TAMOS Agent to the target system’s security settings, see Chapter 3, “Defining Policy,” on page 15.

New features in TAMOS Agent

This section introduces two new TAMOS Agent features, the Native Policy Manager and the TAMOS Agent PAM module, which play important roles in mapping and enforcing policy.

Native Policy Manager

The Native Policy Manager (NPM) is a new daemon that implements the mapping from Tivoli Access Manager for Operating Systems policy to system security settings.

It runs as **pdosnpsd** and its status can be viewed with **pdosctl** using the following command:

```
# pdosctl -s pdosnpsd
```

The daemon is automatically started when the other TAMOS Agent daemons are started and shuts down in a similarly automated fashion.

The NPM has a configuration file in `/var/pdos/etc` called `pdosnpsd.conf`. A template of this file can be found in `/opt/pdos/etc`. This is used to generate the `pdosnpsd.conf` file when the TAMOS Agent is configured.

The NPM employs a Java subprocess to perform most of its core functionality. It uses the default JVM that is automatically installed as part of the TAMOS Agent package.

As updates are made to the Tivoli Access Manager policy, the NPM applies the corresponding changes to the native operating system security settings. If policy updates are made while the NPM is not running, it will apply the changes to the system when it is next started. The NPM receives policy updates through the Tivoli Access Manager policy replication mechanism, so there is some delay between making a policy update and having it applied to the system. To reduce this delay, use the **pdadmin** ‘server replicate’ command.

Policy Reconciliation: Once the policy has been applied to the system, the TAMOS Agent cannot prevent system administrators with sufficient privileges from modifying the system settings, even if this means that the system no longer complies with the established policy. For example, a user with root privileges could extend the maximum password age for a particular user, or update a file without preserving the full set of permissions granted by the Tivoli Access Manager ACL attached in the policy.

To remedy this, the NPM performs a periodic comparison of every element in the policy with the corresponding system settings, reporting on any differences found and reapplying the policy to ensure the system complies. This process is called **reconciliation**.

The **reconcile-period** value in the `pdosnpd.conf` file controls the time period between reconciliation passes. The default time period is 86400 seconds, or one day. The time of day at which the reconciliation process will begin is controlled by the modification timestamp of the file `/var/pdos/rpdb/reconcile.stamp`. The NPM starts the reconciliation process when the difference between the current time and the modification time of the file is greater than the configured reconciliation period. You can adjust the reconciliation schedule at any time by updating the file modification time.

You can also trigger the reconciliation process to run once at any time by creating a file named `/var/pdos/rpdb/reconcile.trigger`. The NPM will perform a reconciliation immediately and then delete the file when the process is complete.

TAMOS Agent PAM module

In order to support enforcement of Tivoli Access Manager for Operating Systems login location, time-of-day restrictions, and holiday policy, the TAMOS Agent includes a new Pluggable Authentication Module (PAM). When the TAMOS Agent is configured, the TAMOS Agent PAM module is added to the system's `/etc/pam.conf` file.

The TAMOS Agent's PAM module is called during login processing. The PAM module retrieves information from the system's PAM infrastructure regarding the login name of the user attempting to login, the terminal, the remote hostname (if the login request is coming from a remote system), and the name of the service over which the login is being attempted.

The TAMOS Agent PAM module uses this information to make a login authorization decision against the defined login location, time-of-day restrictions, and holiday policy. For certain configurations, the use of non-fully qualified domain names may cause logins to be denied more often than expected. For more information on this, refer to "Pluggable Authentication Module Parameters" on page 70.

Note: For the TAMOS Agent's PAM module to be called during login processing, the PAM framework must be enabled on AIX systems.

Supported platforms

As discussed above, the TAMOS Agent product is targeted at operating system platforms which support certain virtualized environments (such as Solaris zones or AIX WPARs) which, for various technical reasons, are not supported by the Tivoli Access Manager for Operating Systems version 6.0 product.

Table 1 presents the operating system configurations supported by the TAMOS Agent. In addition, the table identifies whether or not the Tivoli Access Manager for Operating Systems version 6.0 product is currently supported for that configuration. For operating system configurations which support both the TAMOS Agent and the Tivoli Access Manager for Operating Systems version 6.0 product, only one of those products may be present on the system at any one time.

Table 1. Tivoli Access Manager for Operating Systems Product Support

Platform	TAMOS Agent version 6.0	Tivoli Access Manager for Operating Systems version 6.0
Solaris 10 (without zones)	yes	yes

Table 1. Tivoli Access Manager for Operating Systems Product Support (continued)

Platform	TAMOS Agent version 6.0	Tivoli Access Manager for Operating Systems version 6.0
Solaris 10 (with zones)	yes	no
AIX 6.1 (without WPARs)	yes	yes
AIX 6.1 (with WPARs)	yes	no

Environment prerequisites

The Tivoli Access Manager policy server and the user registry server must be running and network-accessible from the system where the TAMOS Agent is being installed and configured.

The TAMOS Agent is supported in Tivoli Access Manager version 6.0 or later environments.

The TAMOS Agent has the same user registry requirements as Tivoli Access Manager for Operating Systems. That is, the Tivoli Access Manager user registry may be LDAP or Active Directory, and SSL must be used when communicating with the user registry.

For more discussion of TAMOS Agent environments, see “Before you start” on page 5.

Package contents

The TAMOS Agent is distributed as a zip package, `TAMOS-Agent-6.0.0-date-platform.zip`. The platform-specific packages are:

Solaris

`TAMOS-Agent-6.0.0.0-Solaris-SPARC.zip`

This extracts the install files and this PDF document to a directory called `PDOSagent`.

AIX

`TAMOS-Agent-6.0.0.0-AIX.zip`

This extracts the installable package `PDOSagent` and this PDF document.

Chapter 2. Installation and Configuration

This chapter contains the following sections:

- “Before you start”
- “Installation” on page 9
- “Configuration” on page 11
- “Starting the agent” on page 12
- “Stopping the agent” on page 13
- “Unconfiguring the agent” on page 13
- “Uninstalling the agent” on page 14

Before you start

This section details information you will need to know during the installation and configuration processes, including the steps required to enable the Basic Security Module on Solaris platforms.

Note: If the Tivoli Access Manager for Operating Systems product is currently being used on the machine, it must be unconfigured and uninstalled prior to installing the TAMOS Agent.

System Considerations

Before installing, ensure that you:

- Have root permission on the system.
- Have sufficient space available in the /opt and /var file systems. The files associated with the product are installed in the following directories by default:
/opt/pdos
/var/pdos
- Have sufficient space available in the file system where native operating system audit logs will be written. This is important if audit policy (POPs attached to file resources, **AuditAuth**, or **AuditTrace** policy) will be defined. The TAMOS Agent maps this policy to native operating system audit settings. Depending on the defined policy, this could result in the native system generating a large amount of audit data. Follow your operating system’s documented guidelines for setting up auditing.

Required information

The following information will be required when configuring the Tivoli Access Manager Runtime Environment (PDRTE) and the TAMOS Agent:

- The host name of the Tivoli Access Manager policy server, and the port number being used for communications (the default port is 7135).
- The name of the Tivoli Access Manager domain to be used by Tivoli Access Manager for Operating Systems Agent (the default domain is **Default**).
- The name and password of a Tivoli Access Manager administrator.
- The hostname of the system where the Tivoli Access Manager User Registry Server (LDAP server) is running.
- The LDAP server port, if the default port number is not being used (the default port is 389).

- The location of the SSL CA certificate file for the user registry system.
- The user registry suffix to be used by Tivoli Access Manager for Operating Systems.

Considerations when deploying in a Tivoli Access Manager version 6.1 environment

If you have a Tivoli Access Manager version 6.1 environment and you are exploiting the new registry features provided with that version, then you must install the Tivoli Access Manager version 6.1 runtime environment package in place of the Tivoli Access Manager version 6.0 runtime environment package (as referenced below).

Considerations when using Solaris

The TAMOS Agent product supports the global zone and Solaris sparse and whole root zones.

Note: The Tivoli Access Manager for Operating Systems product is not supported on Solaris 10 systems using zones, even in the global zone. The TAMOS Agent can be used in both the global zone and non-global zones if support for TAMOS policy is required on the system.

By default, when a package is installed in a global zone it will also be installed in all non-global zones.

When run in the global zone, the **pkgadd** utility will by default install packages into all defined zones on the system, as well as recording the package to be installed in any zones subsequently created.

If you want to install the TAMOS Agent and its prerequisite software packages to **only** the global zone, then add the **-G** option to the **pkgadd** command.

If you have not installed the packages in the global zone, or if you installed the packages using the **-G** option, then the packages can be installed directly onto a whole root zone by running the **pkgadd** utility from within the whole root zone.

A sparse zone is typically set up with several read-only (inherited) directories which cause problems when installing the TAMOS Agent and its prerequisite software packages directly into a sparse zone. If using sparse zones, the packages should be installed and uninstalled via the global zone using the default options (without the **-G** option)

Table 2 presents the Solaris zone configurations supported by this release of the TAMOS Agent.

Table 2. Package Management

Zone Configuration	Install/Uninstall from Global Zone Only	Install/Uninstall from each Zone independently
Whole Root Zone	yes	yes
Sparse Zone	yes	no

Enabling the Basic Security Module

When using the TAMOS Agent on a Solaris system, operating system auditing must be enabled.

On Solaris systems, if auditing is enabled and a user logs on to the system, the system tracks the identity under which the user logged in. This is known as the **audit id**. The **audit id** does not change even if the user subsequently changes their identity using a command such as the **su** command. By default, the TAMOS Agent requires this **audit id** when making Sudo authorization decisions.

Operating system auditing is enabled by enabling the Solaris Basic Security Module (BSM).

You can see if the BSM is already enabled by running the following command in the global zone:

```
# auditconfig -getcond
```

If the BSM has not been enabled, then this command will fail with an error message such as "Invalid Argument". If BSM has been enabled, then the following message should be generated:

```
audit condition = auditing
```

If the BSM is already enabled, no further steps are required. Otherwise, you must perform the following steps in the global zone as the root user:

1. Run the **bsmconv** command to enable the Solaris Basic Security Module:

```
# /etc/security/bsmconv
```

2. Reboot the system:

```
# shutdown -g 0 -i 6 -y
```

Refer to the Solaris 10 documentation for more details regarding the Basic Security Module.

Auditing requirements on Solaris

In order for the TAMOS Agent to utilize the Solaris auditing subsystem the Solaris Basic Security Module (BSM) must be enabled in the global zone. For details on enabling the Solaris BSM, see "Enabling the Basic Security Module" on page 6.

When using zones, Solaris auditing may be performed either entirely in the global zone or separately in each zone. The TAMOS Agent can work with either of these Solaris audit configurations.

The advantage of controlling Solaris auditing for all zones from within the global zone is that a root user within the non-global zone cannot disable auditing of activities that occur within that zone.

The advantage of controlling Solaris auditing independently for each zone is that the Solaris kernel maintains independent queues for each zone, so resource consumption from one zone cannot impact the operation of another zone.

The default configuration is for Solaris auditing to be performed entirely within the global zone. To enable per-zone auditing issue the following command:

```
auditconfig -setpolicy +perzone
```

To see whether per-zone auditing is already enabled run the command:

```
auditconfig -getpolicy
```

The output should include "perzone". If you enable per-zone auditing, then the audit daemon for running non-global zones needs to be started manually. You do

not need to reboot the non-global zones; the audit daemon can simply be started manually by executing the following command for each non-global zone:

```
/usr/sbin/audit -s
```

This will also ensure that the audit daemon will automatically start the next time the zone is booted. Note that running this command may generate errors even when successful.

Considerations when using AIX

The TAMOS Agent product is supported in the global environment and on AIX System WPARs. AIX Application WPARs are not supported.

Note: The Tivoli Access Manager for Operating Systems product is not supported on AIX 6.1 systems using WPARs, even in the global environment. The TAMOS Agent can be used in both the global environment and system WPARs if support for TAMOS policy is required on the system.

If using AIX System WPARs, the TAMOS Agent (and prerequisite software) must be installed separately into the global environment and on each WPAR where it is to be used. In addition, each System WPAR should be created with a private, writable version of /usr and /opt using the **-l** option to the **mkwpar** command.

If you run the **mkwpar** command without the **-l** option, then /usr and /opt are created as read-only file systems, shared with the global WPAR. This will cause problems when installing and configuring the Tivoli Access Manager Runtime and TAMOS Agent.

Unlike Solaris, in an AIX System WPAR environment, the TAMOS Agent package, and all prerequisite packages, must be individually installed and uninstalled on each of the WPARs within the environment.

Note: When installing into an AIX System WPAR, if the TAMOS Agent and its prerequisite software packages were installed in the global environment prior to the creation of the WPAR, these packages should be uninstalled from the newly created WPAR and then reinstalled.

Table 3 presents the AIX WPAR configurations supported by this release of the TAMOS Agent.

Table 3. WPAR Environments

Creation method	System WPARs	Application WPARs
WPAR Created with -l option to mkwpar	yes	no
WPAR Created without -l option to mkwpar	no	no

AIX Auditing Considerations

For information on AIX auditing requirements, see the AIX Security Expert Audit recommendations page:

http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.security/doc/security/aix_sec_expert_aud_policy_settings.htm

Enabling the Pluggable Authentication Module (PAM) framework on AIX

On AIX, the use of the Pluggable Authentication Module (PAM) framework is not enabled by default.

The PAM framework must be enabled on AIX in order for the TAMOS Agent PAM module to support enforcement of Tivoli Access Manager for Operating Systems login location, time-of-day restrictions, and holiday policy.

To enable the PAM framework, refer to the AIX Information Center:

http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.security/doc/security/pluginauthmod.htm&tocNode=int_324.

Note: When the TAMOS Agent is configured with login policy enabled, the system's `/etc/pam.conf` file is automatically updated with the changes that are required in order to add the TAMOS Agent PAM module. It is not necessary to make these changes manually.

Installation

This section describes how to install the agent, including the steps for extracting and installing prerequisite software.

Installing the prerequisite software

This section describes how to install the prerequisite software on either Solaris or AIX.

Installing on Solaris:

To install the prerequisite software on Solaris:

1. Locate the following packages on the *IBM Tivoli Access Manager for Operating Systems for Solaris* version 6.0 CDs:

- Certificate and SSL Base Runtime (gsk7bas)
- IBM Directory Server Base Client (IDS1bc60)
- IBM Directory Server 32 bit Client (IDS132c60)
- IBM Tivoli Security Utilities (TivSecUtl)
- IBM Tivoli Access Manager License (PDlic)
- IBM Tivoli Access Manager Runtime (PDRTE)

These packages must be installed prior to the TAMOS Agent installation.

2. Install the packages using the Solaris native installation utility **pkgadd**, for example:

```
# pkgadd -d install_image_dir -a install_image_dir/pddefault gsk7bas
# pkgadd -d install_image_dir -a install_image_dir/pddefault IDS1bc60
# pkgadd -d install_image_dir -a install_image_dir/pddefault IDS132c60
# pkgadd -d install_image_dir -a install_image_dir/pddefault TivSecUtl
# pkgadd -d install_image_dir -a install_image_dir/pddefault PDlic
# pkgadd -d install_image_dir -a install_image_dir/pddefault PDRTE
```

3. Check that the packages are installed:

```
# pkginfo | grep -v SUN | grep application
```

application IDS132c60	IBM Directory Server - 32 bit Client
application IDS1bc60	IBM Directory Server - Base Client
application PDRTE	Access Manager Runtime

application PDlic	Access Manager License
application TivSecUtl	Tivoli Security Utilities
application gsk7bas	Certificate and SSL Base Runtime (gsk7bas)

4. PDRTE patch level 3, PDRTE000600-03, or higher is also required. This can be downloaded from the IBM Tivoli support site and installed using native Solaris installation procedures (follow the instructions provided with the patch).

Installing on AIX:

To install the prerequisite software on AIX:

1. Locate the following prerequisite packages on the *IBM Tivoli Access Manager for Operating Systems for AIX* version 6.0 CDs.

- AIX Certificate and SSL Base Runtime (gskta.rte)
- IBM Directory Server client:
 - Client base (No SSL) (idsldap.cltbase60)
 - Client (No SSL) (idsldap.clt32bit60)
 - Client (SSL) (idsldap.clt_max_crypto32bit60)
- Tivoli Security Utilities (TivSec.Utl)
- Tivoli Access Manager License (PD.lic)
- Tivoli Access Manager Runtime (PD.RTE)

These packages must be installed prior to the TAMOS Agent installation.

2. Install the packages using the native AIX installation utility, **installp**, for example:

```
# installp -acgYXd install_image_dir gskta.rte
# installp -acgYXd install_image_dir idsldap.cltbase60
# installp -acgYXd install_image_dir idsldap.clt32bit60
# installp -acgYXd install_image_dir idsldap.clt_max_crypto32bit60
# installp -acgYXd install_image_dir TivSec.Utl
# installp -acgYXd install_image_dir PD.lic
# installp -acgYXd install_image_dir PD.RTE
```

3. PDRTE patch level 3 (included in the Tivoli Access Manager patch 6.0.0-TIV-TAM-FP0003) or higher is also required. This can be downloaded from the IBM Tivoli support site and installed using native operating system installation procedures (follow the instructions provided with the patch).

Extracting the install package

Expand the packaged TAMOS Agent file, TAMOS-Agent-6.0.0-Beta-x-date-platform.zip, into a temporary install directory:

Solaris

```
# cd temp_install_image_dir
# unzip TAMOS-Agent-6.0.0-Beta-x-date-Solaris-SPARC.zip
```

The install files are extracted to a directory called PDOSagent.

AIX

```
# cd temp_install_image_dir
# unzip TAMOS-Agent-6.0.0-Beta-x-date-AIX.zip
```

This extracts the installable package PDOS.agent.

Installing the agent

To install the agent:

Solaris

```
# pkgadd -d temp_install_image_dir PDOSagent
```

AIX

```
# installp -acgYXd temp_install_image_dir PDOS.agent
```

Configuration

This section describes steps for configuring both the Tivoli Access Manager Runtime and the agent. The configuration steps are the same for both Solaris and AIX. The Tivoli Access Manager Runtime and TAMOS Agent must be separately configured in each global and non-global zone or WPAR where the agent will be used.

Configuring the Tivoli Access Manager Runtime

Configure the Tivoli Access Manager Runtime as follows:

```
# pdconfig
```

On the following menus, provide the requested information:

Tivoli Access Manager Setup Menu:

Select 1 (Configure Package).

Tivoli Access Manager Configuration Menu:

Select 1 (Access Manager Runtime Configuration).

Will the policy server be installed on this machine?

Press enter to select default (NO).

Do you want to use Tivoli Common Directory logging?

Press enter to select default (NO).

Registry [1]:

Press enter to accept the default (LDAP).

LDAP server host name:

Enter the hostname where the LDAP server is running.

LDAP server port [389]:

Press enter to accept the default.

Policy server host name:

Enter the hostname where the Tivoli Access Manager Policy Server is running.

Policy server SSL port [7135]:

Press enter to accept the default.

Domain [Default]:

Press enter to accept the default.

Automatically download the pdcert.b64 file from the policy server?

Press enter to accept the default (YES).

The package configuration process runs as follows:

```
The SSL configuration of Access Control Runtime has completed successfully.
Tivoli Access Manager policy server domain name:      Default
Tivoli Access Manager policy server host name:        hostname
Tivoli Access Manager policy server listening port:   7135
The package has been configured successfully.
```

Follow the prompts to exit from **pdconfig**.

Configuring the agent

After successfully configuring the Tivoli Access Manager Runtime, configure the TAMOS Agent as follows:

```
# /opt/pdos/bin/pdoscfg -admin_name sec_master -admin_pwd password -  
registry_ssl_cacert ldap_certificate_name -branch policy_branch_name  
-suffix ldap_suffix
```

The configuration process should run as follows:

```
Determining user registry type.  
Gathering information.  
Processing the current configuration files.  
Processing the command line.  
Validating the information.  
Configuring IBM Tivoli Access Manager for Operating Systems.  
Configuring the PDOSD daemon.  
Initializing the Tivoli Access Manager Policy Server context.  
Registering with Tivoli Access Manager Policy Server. This may take a few minutes.  
Registering the policy-specific policy information.  
...  
Registering the machine-specific policy information.  
Verifying PDOSD daemon's configuration.  
Configuring the PDOSAUDITD daemon.  
Configuring the PDOSWDD daemon.  
The configuration process completed successfully.
```

Notes:

1. Configuration files for the TAMOS Agent are now located under `/var/pdos/etc` rather than `/opt/pdos/etc`. Similarly, the trace configuration files are located under `/var/pdos/trace` rather than `/opt/pdos/etc/trace`. For convenience, symbolic links under the `/opt/pdos/etc` and `/opt/pdos/etc/trace` directories have been created that point to the new locations.
2. **pdoscfg** can be executed multiple times to change an already configured system. For more information about **pdoscfg** and associated command options, see Appendix A, “Command reference,” on page 71.
3. The above **pdoscfg** command shows the minimal set of options required to configure the TAMOS Agent product. There are many other **pdoscfg** options not shown above, which can be used to tailor the TAMOS Agent installation to a particular environment. For more information, see Appendix A, “Command reference,” on page 71.

Starting the agent

To start the agent:

```
# rc.osseal start
```

The agent starts as follows:

```
IBM Tivoli Access Manager for Operating Systems Agent 6.0.0  
pdosd 6.0.0.0 (amos /mnt/amraid/TAMDev/sandboxes/amos/test/dev080331)  
PDOSD is running normally  
Tivoli Access Manager for Operating Systems started successfully
```

Verify the TAMOS Agent Daemons are running:

```
# pdosctl -s
```

Check the response:

```
pdosd is running normally
pdoswdd is running normally
pdoslrd is not running
pdosnpd is running normally
pdosauditd is running normally
```

Stopping the agent

To stop the agent:

```
# rc.osseal stop
```

The agent stops as follows:

```
pdosd shutdown
pdoswdd shutdown
pdosnpd shutdown
pdosauditd shutdown
Tivoli Access Manager for Operating Systems shutdown successfully
```

Unconfiguring the agent

The Tivoli Access Manager for Operating Systems unconfiguration command is **pdosucfg**. This command removes the TAMOS Agent configuration files, disables autostart of the daemons, un-configures the TAMOS Agent PAM module, and un-registers Tivoli Access Manager for Operating Systems with Tivoli Access Manager.

Preparing to unconfigure the TAMOS Agent

Before you unconfigure Tivoli Access Manager for Operating Systems, ensure that:

- The Tivoli Access Manager policy server and the user registry server are running.
- The Tivoli Access Manager Runtime Environment is installed and configured on the same machine as Tivoli Access Manager for Operating Systems.
- You know the Tivoli Access Manager administrator name and administrator password.

Unconfiguring the TAMOS Agent

1. Stop Tivoli Access Manager for Operating Systems.
2. Select one of the following options:
 - a. To unconfigure the TAMOS Agent and **leave the policy branch defined** in the Tivoli Access Manager Policy Server database:

```
pdosucfg -admin_name user_admin_name -admin_pwd user_admin_password
```
 - b. To unconfigure the TAMOS Agent and **remove the policy branch** from the Tivoli Access Manager Policy Server database:

```
pdosucfg -admin_name user_admin_name -admin_pwd user_admin_password
-remove_per_policy on
```

For more information concerning the **pdosucfg** command, see Appendix A, “Command reference,” on page 71.

Unconfiguring the Tivoli Access Manager Runtime

If no other products on the system are using the Tivoli Access Manager Runtime, it can be unconfigured as follows:

```
# pdconfig
```

Follow the menu items to unconfigure the Tivoli Access Manager Runtime.

Uninstalling the agent

This section describes how to uninstall the TAMOS Agent.

If the original TAMOS Agent package was installed independently on each zone (see Table 2 on page 6), the package can be removed from one or more of the system zones independently. If, however, the original TAMOS Agent package was installed on the global zone (without the **pkgadd -G** option), the package should be removed from the global zone (which will automatically remove it from all other system zones). The TAMOS Agent package must be unconfigured and uninstalled from each of the system's WPARs independently.

Notes:

1. Before it can be removed, the TAMOS Agent package must be unconfigured from each zone independently (see "Unconfiguring the agent" on page 13).
2. It is not necessary to reboot the system after uninstalling the TAMOS Agent.

Solaris

To uninstall TAMOS Agent on Solaris:

1. From the command line, login as the root user.

```
# pkgrm PDOSagent
```
2. Confirmation messages are displayed before packages are removed. The order in which they are displayed depends on the order in which the packages are removed. For each package, a confirmation message is displayed:

Do you want to remove this package?

Enter **Yes** and click **Return**.

3. An additional confirmation message is displayed for the runtime package:

This package contains scripts which will be executed with super-user permission during the process of removing this package. Do you want to continue with removal of this package?

Enter **Yes** and click **Return**.

4. When the uninstall process is complete for each package, another confirmation message is displayed:

Removal of package was successful.

AIX

To uninstall TAMOS Agent on AIX:

1. From the command line, login as the root user.

```
# installp -u -g PDOS.agent
```

Chapter 3. Defining Policy

This chapter contains the following sections:

- “Policy administration”
- “Protected Object name structure and access controls” on page 16
- “Protected system resources” on page 16:
 - “Sudo policy” on page 16
 - “Login policy” on page 21
 - “Password management policy” on page 30
 - “File policy” on page 35
 - “Audit policy” on page 48

This chapter explains how to define and set policy for the TAMOS Agent, and details what policy (as defined in the Tivoli Access Manager for Operating Systems environment) is supported by the TAMOS Agent. For this reason, the reader is assumed to be familiar with the structure of policy data in the Tivoli Access Manager for Operating Systems version 6.0 product. For more details see the *Tivoli Access Manager for Operating Systems Administration Guide*.

This chapter also describes whether the TAMOS Agent supports the policy by:

- making authorization decisions against the policy (as happens with the Tivoli Access Manager for Operating System product functionality), or
- mapping the policy to native operating system settings.

The TAMOS agent functions as a user-level endpoint, translating policy that is expressed as Tivoli Access Manager for Operating Systems policy into native operating system security settings. This architectural shift to a user-level endpoint agent has an impact on what existing Tivoli Access Manager for Operating Systems authorization policy is supported by the TAMOS Agent. It also means that once Tivoli Access Manager for Operating Systems policy is mapped to native operating system security settings, the policy will be in effect even if TAMOS Agent daemons are stopped.

If the policy settings are no longer required, then the Tivoli Access Manager for Operating Systems policy should be removed while the TAMOS Agent is running. This will result in the native operating system settings managed by TAMOS Agent being removed.

For a summary of supported Tivoli Access Manager for Operating Systems policy, see “Policy support” on page 57.

Policy administration

Policy administration for the TAMOS Agent is done exactly in the same way as policy administration for the Tivoli Access Manager for Operating System product. A Tivoli Access Manager security administrator can use either Web Portal Manager or the **pdadmin** utility to manage users, groups, and authorization policy in a secure domain.

Please see the *Tivoli Access Manager for Operating System Administration Guide* and the *Tivoli Access Manager Administration Guide* for more information about using the Web Portal Manager or the **pdadmin** utility to define policy.

Protected Object name structure and access controls

The protected object name structure is the same as it is for the Tivoli Access Manager for Operating System product. The root of the object space is /OSSEAL, followed by the policy branch name, the resource type name, and object name.

A TAMOS Agent endpoint can subscribe to multiple policy branches in the same way as a Tivoli Access Manager for Operating System endpoint using the **pdosbranchcfg** command.

Tivoli Access Manager Access Controls Lists (ACLs) and Protected Object Policies (POPs) can be attached to resources defined in the protected object space.

Protected system resources

This section describes protected system resources as they are defined in the Tivoli Access Manager for Operating Systems environment. It specifies what resources can be protected, how the resources are defined in the policy object space, and what actions can be defined for a resource.

This section also describes whether the TAMOS Agent supports the policy by either:

- making authorization decisions against the policy itself (as occurs with the Tivoli Access Manager for Operating System product), or
- mapping the policy to native operating system settings.

Sudo policy

The TAMOS Agent supports enforcement of Sudo policy in the same way as the Tivoli Access Manager for Operating Systems version 6.0 product.

Sudo resources describe commands that require more stringent access control than whether or not a particular program can be executed. Sudo commands allow access control based not only on a command but also on the parameters passed to that command. You can use Sudo commands to remove the requirement for a user to become the root user on a system to perform administrative tasks. Sudo does this by providing the capability to execute a command as a UNIX user other than that of the invoker.

Sudo resources are identified in the Tivoli Access Manager object space in the following way:

`/OSSEAL/policy-branch/Sudo/sudo-command[/sudo-argclass]`

The attributes of the Sudo command are listed in Table 4.

Table 4. Sudo objects

Object name	Description	Type
sudo-command	The name of the Sudo command. This is the object with which parameters describing the actual program, UNIX user identity, and password are associated. You specify this name.	String representing a Sudo command.
sudo-argclass	The name of a class of command arguments. The administrator chooses this name.	String representing a Sudo argument class.

Define the attributes of a Sudo command by creating an object that identifies the Sudo command. Set the Sudo command extended attributes on the object to the appropriate values. Table 5 lists the command attributes.

Table 5. Sudo command attributes

Extended attribute	Description	Type
Sudo-Command	The program to run when access to the Sudo command is granted. This parameter must be specified for Tivoli Access Manager for Operating Systems to consider the Sudo object as valid. This attribute uses a single value.	A fully qualified UNIX file name specifying the program. The string can be a simple UNIX file name, such as /usr/bin/mount, or contain a fixed set of arguments, for example, /usr/bin/rm -i. The arguments specified must be separated by a space and cannot contain any quotation marks.
Sudo-Target-User	The UNIX user name under which the program specified by the Sudo-Command is run. This UNIX user must exist on every system on which the Sudo command needs to run. This attribute is optional. The default value is <i>root</i> . This attribute uses a single value.	A string representing the name of the UNIX user.
Sudo-Invoker-Password	This attribute indicates that the invoker of the Sudo command must enter a password before the command can be executed. The default is to not require the invoker's password. This attribute uses a single value.	The value must be a non-empty string.

Table 5. Sudo command attributes (continued)

Extended attribute	Description	Type
Sudo-Target-Password	This attribute indicates that the invoker of the Sudo command must enter the password of the target user specified by the Sudo-Target-User attribute before the command can be executed. The default is to not require the invoker to supply the target user's password. This attribute uses a single valued.	The value must be a non-empty string.

The execute (x) permission is required to execute a Sudo command as shown in Table 6.

Table 6. Permission required for Sudo

Permission code	Permission name	Permission granted
x	Execute	Execute the Sudo command

Examples of Sudo usage

To define a Sudo command that allows only members of the **sys-admin** group to use the `/usr/sbin/mount` system program and that requires the invoker to enter a password when running the command, you can use the following **pdadmin** commands:

```
pdadmin> object create /OSSEAL/Servers/Sudo/mount "mount" 2 \
    ispolicyattachable yes
pdadmin> object modify /OSSEAL/Servers/Sudo/mount set attribute \
    Sudo-Command /usr/sbin/mount
pdadmin> object modify /OSSEAL/Servers/Sudo/mount set attribute \
    Sudo-Invoker-Password "required"
pdadmin> acl create sudo-mount
pdadmin> acl modify sudo-mount set group sys-admin T[OSSEAL]x
pdadmin> acl attach /OSSEAL/Servers/Sudo/mount sudo-mount
```

You can control what arguments might be provided by defining a Sudo argument class object subordinate to the Sudo command object. Sudo argument class objects are defined in a similar manner to Sudo command objects, by defining extended attributes of the Sudo argument class object.

Table 7 defines an extended attribute that is used to define a Sudo argument class.

Table 7. Extended Sudo attributes for fine-grained control

Extended attribute	Description	Type
Sudo-Arguments	A wildcard string used to match command line arguments. This attribute is multi-valued allowing multiple patterns to describe a single argument class. There is no default value.	A wildcard string used to match command line arguments.

To continue the example, to allow members of the group `net-admin` to mount NFS file systems and only members of the group **`sys-admin`** to mount local file systems use the following **`pdadmin`** commands in addition to the ones above:

```
pdadmin> object create /OSSEAL/Servers/Sudo/mount/remote \
    "Remote mount argument patterns" 0 ispolicyattachable yes
pdadmin> object modify /OSSEAL/Servers/Sudo/mount/remote set attribute \
    Sudo-Arguments "[-]F nfs"
pdadmin> acl create sudo-net-mount
pdadmin> acl modify sudo-net-mount set group net-admin T[OSSEAL]x
pdadmin> acl attach /OSSEAL/Servers/Sudo/mount/remote sudo-net-mount
pdadmin> object create /OSSEAL/Servers/Sudo/mount/local \
    "Local mount argument patterns" 0 ispolicyattachable yes
pdadmin> object modify /OSSEAL/Servers/Sudo/mount/local set \
    attribute Sudo-Arguments "[-]F *"
pdadmin> acl create sudo-local-mount
pdadmin> acl modify sudo-local-mount set group sys-admin T[OSSEAL]x
pdadmin> acl attach /OSSEAL/Servers/Sudo/mount/local sudo-local-mount
pdadmin> acl modify sudo-mount set group sys-admin ""
```

The following notes help explain this example:

- When setting an attribute value in **`pdadmin`**, the value cannot start with a dash character. The dash is represented as `[-]`, a character range containing only a hyphen (-).
- The policy above relies on the precedence of the wildcard patterns. The `[-]F nfs` pattern is more specific than the `[-]F *` pattern.
- The **`sys-admin`** group entry in the **`sudo-mount`** ACL attached to `/OSSEAL/Servers/Sudo/mount` was cleared. This prevents users from accessing the mount Sudo command unless they specify a `-F` parameter as the first option.
- To accommodate the slightly different syntaxes of the mount command on different platforms, you can make the wildcard expressions more complex. For example, mount might expect the `-t` option instead of the `-F` option to specify the file system type, or **`NFS`** might be accepted in place of **`nfs`**. To accommodate two cases, the value of the **`Sudo-Arguments`** attribute of the `/OSSEAL/Servers/Sudo/mount/remote` object can be replaced with `[-][tF][Nn][Ff][Ss]`.
- If the same pattern appears in two different Sudo argument classes of the same Sudo command, warning messages identifying the ambiguous policy are generated in the **`pdosd`** daemon log file and an administrative audit event is generated. The messages do not define which, if any, of the ambiguous policies is applied.

This syntax of UNIX commands can be very complex, allowing specification of command line options and parameters in any order and combination. This can make it difficult to define argument patterns that cover all possibilities. By defining a default behavior that denies access to the Sudo command, the combinations and order of command line options can be restricted to a manageable set.

Wildcards in Sudo arguments

The **`Sudo-Arguments`** attribute uses Tivoli Access Manager for Operating Systems wildcards in a manner similar to the other resource types. The basic elements of the **`Sudo-Arguments`** wildcard strings are the same as the other wildcards with the following exceptions:

- The wildcard asterisk (*) matches a sequence of non-white space characters rather than a sequence of any characters. The asterisk matches an entire command line argument rather than the entire command line. For example, the following pattern matches an arbitrary string as the first argument, followed by the string root as the second argument:

```
* root
```

If the second argument is not root, this pattern does not match.

- A single-space character in the pattern matches any sequence of white-space characters in the string being matched. The special meaning of the space character can be escaped with a backslash character (\), in which case it matches only one space.
- If the Sudo-Arguments attribute has the value "", then this matches the empty string and allows the definition of a pattern that matches when no arguments are passed to the Sudo command.
- The pattern matches a string even if there are trailing arguments that were not matched by the pattern, as long as the preceding arguments matched the entire pattern. For example the pattern:

```
* root
```

Matches both of the strings:

```
show root
add root system
```

Example Sudo policy can be located in the directory /opt/pdos/examples, which contains the following contents:

```
README.samples.sudo
sample.sudo.policy.once
sample.sudo.policy.solaris.per-branch
sample.sudo.policy.aix.per-branch
set_passwd.sh
```

For an explanation of how to use the sample policy files to define Sudo policy, and some examples of using the **pdossudo** utility to exercise this policy, see the file:

```
/opt/pdos/examples/README.samples.sudo
```

How the **pdossudo** command identifies the user executing a Sudo command

The Tivoli Access Manager for Operating Systems product's **pdossudo** command uses the Tivoli Access Manager for Operating Systems accessor ID to identify the user executing a Sudo command when authorizing the operation. This ID is determined by Tivoli Access Manager for Operating Systems kernel module during login and maintained irrespective of any user identity changes that may occur during the user's login session.

Solaris and AIX are both capable of recording a similar ID that stays the same for a user's entire login session, irrespective of any identity changes that may occur during that session. On both systems this is called the **audit ID**.

Solaris requires enabling of the Basic Security Module (BSM) in order for audit IDs to be tracked.

AIX always tracks the audit ID and requires no additional configuration in order to enable this capability.

The TAMOS Agent's **pdossudo** command uses the audit ID to identify the user executing a Sudo command when authorizing that execution

Login policy

The TAMOS Agent supports enforcement of Login time-of-day, holiday, and login location restrictions policy by making an authorization decision during login processing (in much the same way as the existing Tivoli Access Manager for Operating Systems version 6.0 product).

Note: The TAMOS Agent can only support login time-of-day, login holiday, and login location policy for authentication mechanisms set up to use PAM (Pluggable Authentication Modules). This restriction applies on both Solaris and AIX. The TAMOS Agent does not provide an AIX authentication load module.

The TAMOS Agent lets you control when and from where a user can log in to a system. The basic mechanisms for controlling user access are:

- Defining time-of-day login restrictions for users, independent of their log in location, and
- Defining access controls on local and remote terminals.

The TAMOS Agent also provides the ability to define policy related to login activity by mapping defined policy to native operating system settings.

Time-of-day login restrictions

Time-of-day login restrictions are defined by specific policy attributes in the Tivoli Access Manager user registry. They can be specified globally, on a per user basis, or specifically for users unknown to the Tivoli Access Manager runtime (unauthenticated users).

Time-of-day restrictions define hours of the day and days of the week during which users are permitted to log in. For users defined in the Tivoli Access Manager user registry, any user-specific policy overrides any global policy. For users not defined in the Tivoli Access Manager user registry, and, therefore, treated as unauthenticated by Tivoli Access Manager for Operating Systems, the per user policy associated with the special `osseal-unauth` user overrides any global policy.

A time-of-day restriction is defined by a string of the following format:

`day-range:time-range[:utc|:local]`

where:

day-range

Either `anyday`, `weekday`, or a comma-separated list of `sun`, `mon`, `tue`, `wed`, `thu`, `fri`, or `sat`. The `anyday` option indicates that the user is permitted to log in on any day of the week. The `weekday` option specifies that the user is permitted to log in on any day except for Saturday and Sunday. A list of days indicates that the user is permitted to log in only on the specified days.

time-range

Either `anytime` or a start time and end time. The `anytime` option indicates that the user is permitted to log in at any time on the specified days of the week. If time is specified in the form `start_hhmm-end_hhmm`, the `start_hhmm` specifies the hour followed by minutes past the hour for the start time, and the `end_hhmm` specifies the end time.

utc Specifies that the time-of-day restriction should be applied according to Universal Coordinated Time (UTC).

local Specifies that the time-of-day restriction should be applied according to the local time on the system being logged on to. This is the default.

Use the Tivoli Access Manager administration command **pdadmin** to set time-of-day restrictions. The following are examples of time-of-day login policy usage:

- To permit all users to log in only on weekdays from 9:00 A.M. to 5:00 P.M. local time, while permitting the root user to log in at any time, enter:

```
pdadmin> policy set tod-access weekday:0900-1700:local
pdadmin> policy set tod-access anyday:anytime -user root
```
- To additionally constrain unauthenticated users to be allowed to log in only on Mondays, enter:

```
pdadmin> policy set tod-access mon:0900-1700:local -user \ osseal-unauth
```
- To restrict logins regardless of local time zone where the logins take place, enter:

```
pdadmin> policy set tod-access weekday:0900-1700:utc
pdadmin> policy set tod-access anyday:anytime -user root
pdadmin> policy set tod-access mon:0900-1700:utc -user \ osseal-unauth
```

Setting holiday login restrictions

You can specify additional time-of-day restrictions by defining Holidays. Holidays are protected resources that define exceptions to the regular time-of-day restrictions defined in the user registry. The holiday policy is applied when a user logs in. You define a holiday by creating an object with the appropriate attributes set on it. Give the object a name that describes the holiday. The ability of a user to log in on the holiday is controlled by the ACL attached to the same resource. The Login (L) permission must be granted to those users allowed to log in. The format of the value of the Holiday-Dates extended attribute is a start time followed by an optional space and an end time. The specified time format is:

YYYY-MM-DD[-hh[:mm[:ss]]][Z]

Where:

YYYY Year specified as four digits.

MM Month specified as a number from 1 to 12.

DD Day of the month specified as a number from 1 to 31.

hh Hour of the day specified from 0 to 23.

mm Minute of the hour specified from 0 to 59.

ss Second of the minute specified from 0 to 59.

Z Specifies use UTC instead of local time.

The following rules apply when interpreting start and end times that are only partially specified:

- If no end time is specified, the holiday period ends at midnight of the same day on which it started.
- Any time component not specified with a start time defaults to zero.
- If an end time is specified with year, month, and day, but no hour, minute, or second, the holiday period ends at midnight of the day specified.
- If an end time is specified with the hour or the hour and minute, any unspecified components default to zero.
- If either start time or end time are specified in UTC, then their time stamps are interpreted as UTC.

Example of holiday login: Assume that there is a three day holiday around the CEO's birthday, January 18. Only system administrators are allowed to work on January 17, 18, and 19. You could use the following commands to code the Holiday Restriction:

```
pdadmin> object create /OSSEAL/Servers/Login/Holidays/CEO-Birthday-Time "Happy" \
    0 ispolicyattachble yes
pdadmin> object modify /OSSEAL/Servers/Login/Holidays/CEO-Birthday-Time \
    set attribute Holiday-Dates \
    "2001-01-17-09:00:00 2001-01-19-17:00:00"
```

Then create the ACL for the holiday.

```
pdadmin> acl create ceo-birthday-time-acl
pdadmin> acl modify ceo-birthday-time-acl set group sys-admin \
    T[OSSEAL]L
pdadmin> acl attach /OSSEAL/Servers/Login/Holidays/CEO-Birthday-Time \
    ceo-birthday-time-acl
```

This policy permits only members of the Tivoli Access Manager group sys-admin to log in between 9:00 A.M. January 17, 2001 and 5:00 P.M. on January 19, 2001.

Specifying recurring holidays

You can specify recurring holidays by specifying multiple values for the **Holiday-Dates** extended attribute. To define the same CEO-Birthday policy for the year 2002, you can add the following command:

```
pdadmin> object modify /OSSEAL/Servers/Login/Holidays/CEO-Birthday-Time \
    set attribute Holiday-Dates "2002-01-17-09:00:00 2002-01-19-17:00:00"
```

You can also specify holidays that have overlapping ranges. In such cases, the policy that is applied at any time is determined by the following rules:

- The holiday range with the shortest period, if specified, is observed.
- For holiday ranges with the same period, the holiday range with the earliest start time is observed.
- Overlapping ranges specified as multiple values are not combined to form one long range.

To continue the example, if even system administrators were not allowed to log in on the actual birthday of the CEO, January 18, the following holiday policy could be defined:

```
pdadmin> object create /OSSEAL/Servers/Login/Holidays/CEO-Birthday \
    "VeryHappy" 0 ispolicyattachble yes
pdadmin> object modify /OSSEAL/Servers/Login/Holidays/CEO-Birthday \
    set attribute Holiday-Dates "2001-01-18-09:00:00 2001-01-18-17:00:00"
```

With the policy for both the CEO-Birthday holiday and the CEO-Birthday-Time holiday in effect, when a system administrator attempts to log in after 9:00 A.M. on January 17, the time matches the CEO-Birthday-Time holiday range and the login is successful.

If the system administrator attempts to log in after 9:00 A.M. on January 18, the attempt is denied. The shorter time of the CEO-Birthday holiday gives it precedence and the login is not successful.

If you attempt to define multiple holidays with the identical Holiday-Dates attributes, the **pdosd** log file warnings indicate ambiguous policy specifications.

Structure of holiday object names: The structure of the object names beneath the **Holidays** resource type specifier is free form. You can structure the definition of

holiday definitions to use the ACL inheritance. If you define holidays to use ACL inheritance, be aware that the precedence rules carry across all defined holidays within a policy branch without regard to any user-defined hierarchy.

For example, you can define holidays named as:

```
/OSSEAL/policy-branch/Login/Holidays/CEO-Birthday/2001
/OSSEAL/policy-branch/Login/Holidays/CEO-Birthday/2002
/OSSEAL/policy-branch/Login/Holidays/CEO-Birthday/2003
```

with different date ranges specified for each year by attaching separate **Holiday-Dates** attributes for each of the leaf nodes. You could then attach a single ACL to CEO-Birthday.

Login location restrictions

You can specify where users can log in. Define protected resources under the **Terminal** branch of the **Login** resource hierarchy to specify where users can log in.

Note: Login locations are referred to as *terminals* in this document.

Local and remote terminals: Terminals are either:

- **local**, when used for logins to a system from serial devices and graphical consoles, or
- **remote**, when used across a TCP/IP network.

You can group both kinds of terminals together, and use inheritance to define access controls. The names of terminal objects follow the format:

```
/OSSEAL/policy-branch/Login/Terminal/Local/terminal_group/device
/OSSEAL/policy-branch/Login/Terminal/Remote/terminal_group/host
```

Where:

terminal_group

A string for an administrator-definable logical grouping of terminals that allow the application of inherited access controls. This part of the object name must be included.

device The fully qualified UNIX file name of a terminal device on a system. For example, `/dev/console` or `/dev/tty/0`. This file name cannot include wildcard characters.

host The representation of a host, group of hosts, or network. One of the following:

- A fully qualified host name from `/etc/hosts`, DNS, etc. The name can include wildcard characters such as the asterisk (*) or the question mark (?) but must always represent a fully qualified name. A simple, short name is not allowed.
- An IP address-netmask combination in dot notation (*IPv4_address:number_of_bits*). The netmask (*:number_of_bits*) is optional. The absence of a specified netmask implies a 32-bit netmask, that is, a host address.
- An IPv6 address-address mask combination in standard IPv6 text representation, enclosed within square brackets ([]) (*[ipv6_address]:[IPv6_netmask]*). The address mask (*: [IPv6_netmask]*) is optional. The absence of a specified IPv6 mask implies a 128-bit mask.

The following list contains examples of login resource specifications:

- `/OSSEAL/policy-branch/Login/Terminal/Local/Modems/dev/tty063`

- /OSSEAL/policy-branch/Login/Terminal/Remote/Development/*.dev.company.com
- /OSSEAL/policy-branch/Login/Terminal/Remote/Xterms/10.1.34.2:24 v
/OSSEAL/policy-branch/Login/Terminal/Remote/IPv6Terms/
[ABCD:0:0:FEDC:0:0:8813:830A]

Access control on Login objects: Attach Tivoli Access Manager access controls at the appropriate point in the Login/Terminal object hierarchy to control resources. For example, you can establish a default policy that controls system access from remote locations by attaching an ACL or a POP to the /OSSEAL/policy-branch/ Terminal/Remote object. Table 8 shows the permission required.

Table 8. Valid permission to log in

Permission name	Description
Login (L)	Permission from the associated terminal

Uniqueness of terminal resources definitions: A terminal can appear in only one terminal group within each policy branch. If a terminal appears in more than one group, a warning is generated in the **pdosd** error log. The prevailing policy authorization is undefined. You might not get the expected results.

Login activity policy

The TAMOS Agent supports the ability to define policy related to login activity by mapping defined policy to native operating system settings. The TAMOS Agent does not enforce this policy directly, as was done with the Tivoli Access Manager for Operating Systems product.

The policy is defined centrally by using extended attributes of the /OSSEAL/policy-branch/Login object and controls the following aspects of login activity:

- Password expiry
- Account suspensions due to failed login attempts
- Account lockouts due to account inactivity

User exception policy allows defining exceptions to the default login activity policy for a specific UNIX user name. It is defined by attaching the extended attributes to: /OSSEAL/policy-branch/Login/UserExceptions/unix_name

Support for extended attributes that control login activity policy depends on the capabilities provided by the native operating system. The following sections describe which extended attributes are supported and how they are mapped to native operating system settings for Solaris and AIX.

Tivoli Access Manager for Operating Systems login activity policy supported on Solaris: Table 9 describes the extended attributes that control login activity policy and are supported on Solaris.

Table 9. Login activity policy attributes supported on Solaris

Login activity attribute	Description	Type
Login-MinPasswordDays	Minimum amount of time before a password can be changed. If not specified, a default value of zero is used. A value of zero indicates the password can be changed as frequently as a user likes.	Non-negative integer

Table 9. Login activity policy attributes supported on Solaris (continued)

Login activity attribute	Description	Type
Login-MaxPasswordDays	Maximum amount of time before a password must be changed. If not specified, the default value of zero is assumed and the passwords will never be considered to have expired.	Non-negative integer
Login-MaxInactiveDays	The number of days before an inactive account is locked permanently. An account is considered inactive from the last time a successful login occurred to that account. If not specified, the default value of zero is assumed and inactive accounts are never locked.	Non-negative integer
Login-MaxFailedLogins	Number of failed login attempts on an account before that account is suspended. If the Login-MaxFailedLogins attribute is not specified, the default value of zero is assumed and the account will never be suspended due to failed login attempts.	Non-negative integer
Login-PolicyDisabled	Used to disable all login activity policy. If this attribute is present and the value is a non-empty string, then none of the defined login activity policy is enforced.	Non-empty string

The following Tivoli Access Manager for Operating Systems login activity extended attributes are not supported for the Solaris platform and are not mapped to native Solaris settings:

- **Login-LockMinutes**,
- **Login-LoginMinutes**,
- **Login-MaxGraceLogins**, and
- **Login-MaxConcurrent**.

Mapping login activity attributes to native Solaris settings: The TAMOS Agent maps the Tivoli Access Manager for Operating Systems policy login activity extended attributes on a Solaris machine by modifying settings in the following native Solaris files:

```
/etc/shadow
/etc/default/login
/etc/security/policy.conf
/etc/user_attr
```

The mapping is done in the following way: If no login activity policy attributes are defined, and there are no user exceptions defined, then TAMOS Agent will not make any modifications to the system settings to apply login activity policy. If at least one login activity policy attribute is defined, or any user exception objects are defined, then TAMOS Agent will apply all of the login activity policy settings to all users on the system using the default values shown in Table 9 on page 25.

When determining the effective value of a login activity policy attribute for a user, TAMOS Agent first checks for a user exception, then checks the default login

activity policy, and finally falls back to using default values. This is different to the behavior of the Tivoli Access Manager for Operating Systems product, which only applies the default login activity policy if there are no user exceptions for the user

The **Login-minPasswordDays**, **Login-maxPasswordDays**, and the **maxInactiveDays** are mapped directly to the **min**, **max**, and **inactive** fields in the native `/etc/shadow` file. If attributes are defined for the `/OSSEAL/policybranch/` Login object as the default policy, the TAMOS Agent updates every user entry in the `/etc/shadow` file. If these attributes are defined as **UserException** policy, then for each specified user, the corresponding entry in the native Solaris `/etc/shadow` file is updated accordingly.

One side effect of defining the **minPasswordDays** and **maxPasswordDays** attributes for the `/OSSEAL/policybranch/` Login object as the default policy is that the TAMOS Agent clears the **MINWEEKS** and **MAXWEEKS** fields in the `/etc/default/passwd` file.

The **Login-MaxFailedLogins** attribute does not map directly to a single Solaris attribute setting. To map this attribute, the following Solaris attributes are manipulated, depending on the defined policy:

- the **RETRIES** field in the `/etc/default/login` file,
- the **LOCK_AFTER_RETRIES** field in the `/etc/security/policy.conf` file, and
- the **lock_after_retries** attribute in the user stanzas in `/etc/user_attr`.

The **RETRIES** field in `/etc/default/login` is set to the default **Login-MaxFailedLogins** value if nonzero, otherwise, if any **UserException** policy has been defined with nonzero values for **Login-MaxFailedLogins**, the minimum nonzero value is used.

The **LOCK_AFTER_RETRIES** field in `/etc/security/policy.conf` is set to *yes* if the default **Login-MaxFailedLogins** value is nonzero, otherwise, it is set to *no*.

The **lock_after_retries** attribute in individual user stanzas in the `/etc/user_attr` file is set under the following conditions:

- if the default **Login-MaxFailedLogins** is not set, or is set to zero, for users with **UserException** policy that includes a nonzero value for **Login-MaxFailedLogins**, the **lock_after_retries** is set to *yes*.
- if the default **Login-MaxFailedLogins** is nonzero, for users with **UserException** policy that includes a zero value for **Login-MaxFailedLogins**, the **lock_after_retries** is set to *no*.
- for the root user, the **lock_after_retries** value is left as the default value, *no*, unless **UserException** policy is defined for the root user that sets **Login-MaxFailedLogins** to a nonzero value.

If the **Login-MaxFailedLogins** attribute is set as the default policy, the TAMOS Agent sets the **RETRIES** field to the specified value and the **LOCK_AFTER_RETRIES** field to *yes*.

Mapping **Login-MaxFailedLogins** attribute values defined in **UserException** policy is more complicated because Solaris does not provide a way to specify the number of retries on a per-user basis (you can only specify whether or not the default **RETRIES** value applies).

If, in addition to the default policy, the **Login-MaxFailedLogins** attribute is defined in **UserException** policy with a non-zero value, this policy will have no effect.

If the **Login-MaxFailedLogins** attribute is defined in **UserException** policy with a zero value, then for each specified user, the **lock_after_retries** field in the corresponding user entry in the native Solaris `/etc/user_attr` file is set to *no*, which will override the default policy.

If the **Login-MaxFailedLogins** attribute is not defined as the default policy or is defined with a zero value, but the attribute is defined with a non-zero value as part of **UserException** policy, then for each specified user, the **lock_after_retries** field in the corresponding user entry in the native Solaris `/etc/user_attr` file is set to *yes* and the **RETRIES** field is set to the minimum value of all the **Login-MaxFailedLogins** values specified in the **UserException** policy.

If the **Login-MaxFailedLogins** attribute is not defined as the default policy, or is defined with a zero value, but the attribute is only defined with a zero value as part of **UserException** policy, the TAMOS Agent sets the **RETRIES** field to the specified value in the native Solaris `/etc/default/login` file and the **LOCK_AFTER_RETRIES** field to *no* in the `/etc/security/policy.conf` file.

Note: By default on a Solaris system, the root user entry in the `/etc/user_attr` file always has the **lock_after_retries** field set to *no*, which means that the defined **RETRIES** value will not effect the root user. Presumably, the intention is to avoid locking the root user out of the system. Be aware, however, that if **UserException** policy is defined for the root user and the **Login-MaxFailedLogins** attribute is set to a non-zero value, the TAMOS Agent will change the root user's **lock_after_retries** attribute to *yes*, which may result in the root user getting locked out of the system.

Tivoli Access Manager for Operating Systems login activity policy supported on AIX: Table 10 describes the extended attributes that control login activity policy and are supported on AIX.

Table 10. Password management policy attributes supported on AIX

Login activity attribute	Description	Type
Login-MinPasswordDays	Minimum amount of time before a password can be changed. If not specified, a default value of zero is used. A value of zero indicates the password can be changed as frequently as a user likes.	Non-negative integer
Login-MaxPasswordDays	Maximum amount of time before a password must be changed. If not specified, the default value of zero is assumed and the passwords will never be considered to have expired.	Non-negative integer
Login-MaxFailedLogins	Number of failed login attempts on an account before that account is suspended. If the Login-MaxFailedLogins attribute is not specified, the default value of zero is assumed and the account will never be suspended due to failed login attempts.	Non-negative integer

Table 10. Password management policy attributes supported on AIX (continued)

Login activity attribute	Description	Type
Login-PolicyDisabled	Used to disable all login activity policy. If this attribute is present and the value is a non-empty string, then none of the defined login activity policy is enforced.	Non-empty string

The following Tivoli Access Manager for Operating Systems login activity extended attributes are not supported for the AIX platform and are not mapped to native AIX settings:

- Login-LockMinutes,
- Login-LoginMinutes,
- Login-MaxGraceLogins,
- Login-MaxConcurrent, and
- Login-MaxInactiveDays.

Mapping login activity attributes to native AIX settings

The TAMOS Agent maps the Tivoli Access Manager for Operating Systems policy login activity extended attributes on a AIX machine. All the supported login activity attribute values are set in the AIX `/etc/security/user` file. Global policy is set in the "default" stanza. User exceptions are set in the user's stanza.

On AIX, per-user configuration is done in the `/etc/security/user` file. Each user has a stanza in this file where the various values can be set. There is also a default stanza that applies if no values are set for the user.

Table 11. AIX per-user configuration

Login Activity Policy Attribute	/etc/security/user field
Login-MinPasswordDays	minage *
Login-MaxPasswordDays	maxage *
Login-MaxFailedLogins	loginretries

* Since the Tivoli Access Manager for Operating Systems policy is defined in days and the AIX policy is defined in weeks, the values for these attributes need to be converted from days to weeks. This is done by dividing the number of days by 7 and rounding down, with a minimum of 1.

If no login activity policy attributes are defined, and there are no user exceptions defined, then TAMOS Agent will not make any modifications to the system to apply login activity policy. If at least one login activity policy attribute is defined, or any user exception objects are defined, then TAMOS Agent will apply all of the login activity policy settings on the system using the default values shown in Table 10 on page 28.

When determining the effective value of a login activity policy attribute for a user, TAMOS Agent first checks for a user exception, then checks the default login activity policy, and finally falls back to using default values. This is different to the behavior of the Tivoli Access Manager for Operating Systems product, which only applies the default login activity policy if there are no user exceptions for the user.

The **Login-MinPasswordDays**, **Login-MaxPasswordDays**, and the **Login-MaxFailedLogins** attributes are defined for the */OSSEAL/policybranch/Login* object as the default policy. The TAMOS Agent sets the corresponding AIX attributes in the default stanza of the native AIX */etc/security/user* file. If these attributes are defined as **UserException** policy, then for each specified user, the corresponding attributes in the user's stanza of the native AIX */etc/security/user* file is updated accordingly.

For example, if the following policy is defined:

- */OSSEAL/branch/Login*
Extended attribute **Login-MinPasswordDays** is defined with a value of 35 days, and **Login-MaxPasswordDays** is defined with a value of 49 days.
- */OSSEAL/branch/Login/UserExceptions/bob*
Extended attribute **Login-MinPasswordDays** is defined with a value of 21 days, and **Login-MaxPasswordDays** is defined with a value of 42 days.

The resulting mapping to the */ec/security/user* file results in two stanzas being updated, 'default' and 'bob' that contain the **minage** and **maxage** attributes, which are set as shown:

```
default:
minage = 5
maxage = 7
```

```
bob:
minage = 3
maxage = 6
```

Password management policy

The TAMOS Agent supports Tivoli Access Manager for Operating Systems password management policy by mapping defined policy to native operating system settings. The capabilities provided by the target platform determine which attributes can be successfully mapped. The TAMOS Agent does not enforce this policy directly, as was done with the Tivoli Access Manager for Operating Systems product.

The policy is defined centrally by using password management extended attributes, and controls the following aspects of password management:

- Password strength
- Password aging

Default policy is defined by attaching the required Password management extended attributes to */OSSEAL/policy-branch/Password* object. User exception policy allows defining exceptions to the default password management policy for a specific UNIX user name. It is defined by attaching the extended attributes to: */OSSEAL/policy-branch/Password/UserExceptions/unix_name*

Support for extended attributes that control password management policy depends on the capabilities provided by the native operating system. The following sections describe which extended attributes are supported and how they are mapped to native operating settings for Solaris and AIX .

Tivoli Access Manager for Operating Systems password management policy supported on Solaris

Table 12 on page 31 describes the extended attributes that control password management policy that are supported on Solaris.

Table 12. Password management policy attributes supported on Solaris

Password management attribute	Description	Type
Password-MinPasswordLen	Minimum allowed length of a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordAlpha	Minimum number of alphabetic characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordNumeric	Minimum number of numeric characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordLower	Minimum number of lower case characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordUpper	Minimum number of upper case characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordSpecial	Minimum number of special characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordDays	Minimum amount of time before a password can be changed. If a not specified, a default value of zero is used, indicating that the password can be changed as frequently as a user wants.	Non-negative integer
Password-MaxPasswordRepeat	Maximum number of times a character can be repeated consecutively in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-PasswordMaxConsPrev	If set, this value is the maximum number of consecutive characters that can be in common between the new password and the old password.	Non-negative integer

Table 12. Password management policy attributes supported on Solaris (continued)

Password management attribute	Description	Type
Password-PasswordHistory	The number of passwords to store as a history. When a new password is set, it cannot be one of these previous passwords. A limited number of history elements will be supported. The maximum value of the password history is 10. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-PasswordNameCheck	Check that the password is not contained in and does not contain the user ID. If not specified, a default value of zero is used, indicating that the user ID can be used in the password.	Non-negative integer

The following Tivoli Access Manager for Operating Systems password management attributes are not supported for the Solaris platform and are not mapped to native Solaris settings:

- **Password-MinPasswordAlphaNum**,
- **Password-PasswordOldPwdCheck**, and
- **Password-PasswordNonNumFirstLast**.

Password policy mapping for Solaris

All the supported password management attribute values are mapped to the native Solaris `/etc/default/passwd` file (except for **Password-minPasswordDays**), which means that only global password policy can be applied for most password policy attributes.

The **Password-minPasswordDays** attribute is applied in the `/etc/shadow` file in the same way the **Login-minPasswordDays** attribute is applied. The **Password-minPasswordDays** attribute has a higher precedence over the **Login-minPasswordDays** attribute

Settings applied in `/etc/default/passwd` are shown in Table 13:

Table 13. Password Management Policy Attribute settings

Password Management Policy Attribute	/etc/default/passwd field
Password-MinPasswordLen	PASSLENGTH
Password-MinPasswordAlpha	MINALPHA
Password-MinPasswordNumeric	MINDIGIT
Password-MinPasswordLower	MINLOWER
Password-MinPasswordUpper	MINUPPER
Password-MinPasswordSpecial	MINSPECIAL
Password-MaxPasswordRepeat	MAXREPEATS
Password-PasswordMaxConsPrev	MINDIFF1 *
Password-PasswordHistory	HISTORY

Table 13. Password Management Policy Attribute settings (continued)

Password Management Policy Attribute	/etc/default/passwd field
Password-PasswordNameCheck	NAMECHECK

* MINDIFF will be set to the maximum of 0 and **minPasswordLength** - **maxConsecutivePrevious**. If **minPasswordLength** is not set, MINDIFF will be set to 0. This is the only case where the value from the policy needs to be transformed.

If no password management policy attributes are defined, and there are no user exceptions defined, then TAMOS Agent will not make any modifications to the system to apply password management policy. If at least one password management policy attribute is defined, or any user exception objects are defined, then TAMOS Agent will apply all of the password management policy settings to all users on the system using the default values shown in Table 12 on page 31.

When determining the effective value of a password management policy attribute for a user, TAMOS Agent first checks for a user exception, then checks the default login activity policy, and finally falls back to using default values. This is different to the behavior of the Tivoli Access Manager for Operating Systems product, which only applies the default password management policy if there are no user exceptions for the user.

Note: On Solaris, privileged users (for instance, real and effective uid equal to 0) are not forced to comply with password aging and password construction requirements. The password policy management settings do not apply to privileged users, whether they are changing their own password or another user's password.

Tivoli Access Manager for Operating Systems password management policy supported on AIX

Table 14 describes the extended attributes that control password management policy and are supported on AIX.

Table 14. Login activity policy attributes supported on AIX

Password management attribute	Description	Type
Password-MinPasswordLen	Minimum allowed length of a password. If not specified, a default value of zero is used, indicating that there is no restriction	Non-negative integer
Password-MinPasswordAlpha	Minimum number of alphabetic characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-MinPasswordSpecial	Minimum number of special characters required in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer

Table 14. Login activity policy attributes supported on AIX (continued)

Password management attribute	Description	Type
Password-MinPasswordDays	Minimum amount of time before a password can be changed. If a not specified, a default value of zero is used, indicating that the password can be changed as frequently as a user wants.	Non-negative integer
Password-MinPasswordRepeat	Maximum number of times a character can be repeated consecutively in a password. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer
Password-PasswordMaxConsPrev	If set, this value is the maximum number of consecutive characters that can be in common between the new password and the old password.	Non-negative integer
Password-PasswordHistory	The number of passwords to store as a history. When a new password is set, it cannot be one of these previous passwords. A limited number of history elements will be supported. The maximum value of the password history is 10. If not specified, a default value of zero is used, indicating that there is no restriction.	Non-negative integer

The following Tivoli Access Manager for Operating Systems password management attributes are not supported for the AIX platform and are not mapped to native AIX settings:

- **Password-MinPasswordAlphaNum,**
- **Password-MinPasswordNumeric,**
- **Password-MinPasswordLower,**
- **Password-MinPasswordUpper,**
- **Password-PasswordNameCheck,**
- **Password-PasswordOldPwdCheck,**
- **Password-PasswordNonNumFirstLast.**

Password policy mapping for AIX

All the supported password management attribute values are set in the AIX /etc/security/user file. Global policy is set in the default stanza. Per-user overrides are set in the user's stanza.

If no password management policy attributes are defined, and there are no user exceptions defined, then TAMOS Agent will not make any modifications to the system to apply password management policy. If at least one password management policy attribute is defined, or any user exception objects are defined, then TAMOS Agent will apply all of the password management policy settings on the system using the default values shown in Table 15 on page 35.

When determining the effective value of a password management policy attribute for a user, TAMOS Agent first checks for a user exception, then checks the default login activity policy, and finally falls back to using default values. This is different to the behavior of the Tivoli Access Manager for Operating Systems product, which only applies the default password management policy if there are no user exceptions for the use

Table 15 shows the settings applied in `/etc/security/user`:

Table 15. Password Management Policy Attribute settings

Password Management Policy Attribute	/etc/security/user field
Password-MinPasswordLen	minlen
Password-MinPasswordAlpha	minalpha
Password-MinPasswordSpecial	minother *
Password-MinPasswordDays	minage * *
Password-MaxPasswordRepeat	maxrepeats
Password-PasswordMaxConsPrev	mindiff * * *
Password-PasswordHistory	histsize

* **minother** includes numeric characters, whereas **minPasswordSpecial** does not. But in order to allow **minother** to be controlled by Tivoli Access Manager for Operating Systems policy, this is the best fit.

* * **minage** is measured in weeks. This is done by dividing the number of days by 7 and rounding down, with a minimum of 1. A value set here overrides any value set in the Login Activity policy section (as described above).

* * * **mindiff** will be set to the maximum of 0 and **minPasswordLength - maxConsecutivePrevious**. If **minPasswordLength** is not set, **mindiff** will be set to 0. This is the only case where the value from the policy needs to be transformed.

File policy

TAMOS Agent supports the Tivoli Access Manager for Operating Systems File policy by mapping the defined policy to native file system access controls in the form of extended ACLs. The TAMOS Agent does not enforce this policy directly, as was done with the Tivoli Access Manager for Operating Systems product.

Management of file protection policy is implemented by attaching Tivoli Access Manager ACLs to objects that are defined below the **File** resource under a Tivoli Access Manager for Operating Systems policy branch.

File system resources are represented in the Tivoli Access Manager object space by defining an object name with resource type **File** and specifying the name of the file system resource to be protected:

`/OSSEAL/policy-branch/File/filespec`

Table 16 details the file system objects.

Table 16. File system objects

Object name	Description	Type
Filespec	An object name that represents a file system resource. The string specifies the full path name of the file resource being protected.	String conforming to the UNIX file-naming rules.

Some example file system resource specifications are:

```
/OSSEAL/Default/File/etc/passwd  
/OSSEAL/Default/File/var/appA/*.log
```

Tivoli Access Manager ACLs can be attached to file system resources represented in the object space.

Table 17 details the valid ACL permissions that can be associated with File system resources.

Table 17. File permissions

	Permission name (OSSEAL actions)	Permission granted
For files:	Read (r)	Access a file for reading.
	Write (w)	Access a file for writing.
	Execute (x)	Create a particular file system resource.
For directories:	Create (N)	Execute a file.
	Chown (o)	Change the ownership of a file system resource.
	Chdir (d)	Change directory into a file system directory resource.
	Rename (R)	Move (or rename) a file system resource.
	Delete (d)	Remove a file system resource.
	Utime (u)	Modify the file access and modification times associated with a file system resource.
	List (l)	List the contents of a directory.
Tivoli Access Manager base action:		
For directories:	Traverse (T)	Grants permission to pass through a container object (only applies to directory resources).

A Tivoli Access Manager ACL will contain entries for users and groups. These entries will be mapped to user and group entries in a native file system extended access control list. The any-other ACL entry, if it is defined, gets mapped to the other (or world) entry in the standard UNIX permissions associated with file

resources. The unauthenticated entry of the ACL is ignored. With just the basic **File** policy defined, no changes will be made to the native file resource's owning user and the owning user's UNIX permissions, the owning group and the owning group's UNIX permissions, or the **setuid**, **setgid**, or **sticky bit** in the UNIX permissions. To allow for more control over these aspects of the native file resource, new policy has been introduced for use with the TAMOS Agent.

New policy for File resources

To define policy to manage the owning user's and owning group's UNIX permissions, create a special Tivoli Access Manager user ACL entry for the user name **owning-user** and a special Tivoli Access Manager ACL group entry for the group name **owning-group**. The TAMOS Agent will map the permissions defined in these entries to the owning user's and the owning group's UNIX permissions. Absence of these special user and group ACL entries indicates that the corresponding UNIX permission set is not managed by TAMOS Agent.

Note: The **owning-user** and **owning-group** identities are automatically created in the Tivoli Access Manager user registry as part of the TAMOS Agent installation process.

To define policy to manage the actual owning user name and the owning group name of the native file system resource, two new extended attributes **owning-user** and **owning-group** have been introduced. The new extended attributes can be defined in either the attached ACL that is used to protect the file resource object or on the protected object itself. Extended attributes specified directly on the Tivoli Access Manager protected object take precedence over those specified in the attached ACL.

Note: If the **owning-user** or **owning-group** extended attribute conveys an invalid or nonexistent user or group name, ownership of the **filesystem** object is left unchanged.

To define policy to control the UNIX special file permissions, **setuid**, **setgid**, and the **sticky bit**, a new extended attribute called **special-bits** has been introduced. Unlike the **owning-user** and **owning-group** extended attributes, the **special-bits** extended attribute may only be applied on the protected object (not on the attached ACL). The value of the **special-bits** extended attribute is a comma-delimited list, with the desired attribute specified strictly in order with no spaces. In order to set the **setuid** bit, the string **setuid** must be specified, followed by the text string **setgid** (if that is to be set), followed by the text string **sticky** (if the sticky bit is to be set). If the attribute value does not follow this format, or it includes additional bits, no special permissions will be applied to the file and a warning will be logged.

If no special permissions should be applied to the file, the attribute should be set to *none*. If the **special-bits** extended attribute is not applied to the File resource protected object, then the file's special permissions will be left untouched.

The new extended attributes are shown in Table 18.

Table 18. New extended attributes for File policy

Extended Attribute	Description	Defined on
owning-user	User name to map as the file system resource's owning user.	The protected object name associated with the file resource or the Tivoli Access Manager ACL attached to the protected object name.
owning-group	Group name to map as the file system resource's owning group.	The protected object name associated with the file resource or the Tivoli Access Manager ACL attached to the protected object name.
special-bits	Text string specifying whether or not the setuid , setgid , and sticky bit should be set in the file's UNIX permission set.	The protected object name associated with the file resource.

The following restrictions apply when defining File policy:

- The OSSEAL K (kill) permission is not supported, since it cannot be mapped to a UNIX permission.
- The file system hosting the file or directory being protected must support one of the following types of extended file system ACLs:
 - POSIX style ACLs,
 - NFSv4 ACLs, or
 - AIX Classic ACLs.
- ACLs are not inherited. This means that when an ACL is attached to a directory it is only the directory's permissions that are controlled by the TAMOS Agent and not the permissions of any files or subdirectories contained in the directory.
- Tivoli Access Manager for Operating Systems **Access Restrictions** extended attributes are not supported.
- Wildcard specification in a File protected object name is only supported for file resources and only applies to the immediate children within the directory under which the file resource name containing wildcards is specified. ACLs attached to file resources specified using wildcards are supported. POPs attached to file resources specified using wildcards are not supported.

Mapping Tivoli Access Manager for Operating Systems File policy to native file system Access Control settings

The TAMOS Agent maps Tivoli Access Manager ACLs that are directly attached to file system resources (represented in the Tivoli Access Manager object space under `/OSSEAL/policy_branch/File`) to the target operating system's file access control mechanism. The mapping is based on the type of extended file system ACLs supported by the file system where the protected file or directory resource resides.

The TAMOS Agent supports mapping File policy to file systems that support the following types of extended file system ACLs:

- POSIX style ACLs (Solaris UFS file system),
- NFSv4 ACLs (Solaris ZFS file system and AIX JFS2 file systems with extended attribute version 2 support), or
- AIX Classic ACLs (AIX JFS and JFS2 file systems).

The following sections show the mappings for these three types of extended ACLs.

Support for file systems with POSIX style ACLs (Solaris UFS filesystem)

For file systems with POSIX style ACLs, such as the Solaris UFS filesystem, the TAMOS Agent maps OSSEAL actions and Tivoli Access Manager base actions to UNIX file system permissions as follows:

Table 19. Mapping OSSEAL actions to UNIX permissions

OSSEAL actions		UNIX permissions
For files:	r (read)	r
	w (write)	w
	x (execute)	x
For directories:	l (list directory)	r
	N (create), o (chown), p (chmod), R (rename), d (delete), U (utime)	w
	D (chdir)	x
Tivoli Access Manager base action:		
For directories:	T (traverse)	x

Note: All other OSSEAL actions and Tivoli Access Manager base actions are ignored.

Tivoli Access Manager ACL entry mapping: Each ACL entry in a Tivoli Access Manager ACL maps to an ACL entry in the corresponding file or directory's POSIX extended ACL.

Any user or group referenced in a Tivoli Access Manager ACL entry must be defined on the target system in order for the corresponding ACL entry to take effect. If a user or group referenced in an ACL entry is not defined on the target system, then that ACL entry is ignored. In that case, other ACL entries for which the user or group is defined on the system are still applied.

A log message is generated on the target system for any user or group not defined on that system, indicating that policy cannot be completely mapped to a POSIX extended ACL. The any-other entry maps to the file or directory's world access permissions. For example, consider the following Tivoli Access Manager ACL:

```
group dba T[OSSEAL]rwxlN
group dbread T[OSSEAL]rxl
any-other T
```

When attached to a file system resource representing a file, this Tivoli Access Manager ACL would set entries in the extended ACL associated with the file for the groups **dba** and **dbread**, as shown.

The following output corresponds to that shown by the Solaris `ls -v` command:

```
-rwxr-x---+ 1 root    root    some-file
0:user::rwx
1:group::r-x
```

```
2:group:dba:rwx
3:group:dbread:r-x
4:mask:rwx
5:other:---
```

When attached to a file system resource representing a directory, the T (traverse) permission is also mapped, and the extended ACL associated with the directory would have the following entries.

The following output corresponds to that shown by the Solaris `ls -v` command:

```
-rwxr-x--x+  1 root    root    some-directory
```

```
0:user::rwx
1:group::r-x

2:group:dba:rwx
3:group:dbread:r-x
4:mask:rwx
5:other:--x
```

As described above, Tivoli Access Manager ACL entries only map to POSIX extended ACL entries and not to the owning user or owning group permissions. This minimizes the risk that permissions, which are relied on by the system service or application to which the file belongs, will change access control in a way that inadvertently disrupts system or application operation. In order to manage the owning user or owning group permissions, you can specify a predefined user (**owning-user**) or group (**owning-group**) ACL entry.

For example, if the Tivoli Access Manager ACL shown above is amended to include an owning-user and owning-group entry as follows:

```
user owning-user [OSSEAL]rwx
group owning-group [OSSEAL]rx
group dba [OSSEAL]rwx
group dbread [OSSEAL]rx
any-other -
```

Then the user and group entries of the file's extended ACL are now managed by the TAMOS Agent:

The following output corresponds to that shown by the Solaris `ls -v` command:

```
-rwxr-x---+  1 root    root    some-file
```

```
0:user::rwx-
1:group::r-x
2:group:dba:rwx
3:group:dbread:r-x
4:mask:rwx
5:other:---
```

Note: The above example is for illustrative purposes only. It may not be wise to remove the x permission for the owning user and owning group, depending on the actual file in question.

If you also want to manage the name of the owning user and owning group of the file or directory in TAMOS Agent policy (to ensure that it doesn't get changed) then you can specify the following extended attributes, either in the attached ACL that is used to protect the object or on the Tivoli Access Manager protected object itself. Extended attributes specified directly on the Tivoli Access Manager protected object take precedence over those specified in the attached ACL.

The attributes are defined in Table 18 on page 38.

So if the example Tivoli Access Manager ACL above is amended as follows:

```
user owning-user [OSSEAL]rwx
group owning-group [OSSEAL]rx
group dba [OSSEAL]rwx
group dbread [OSSEAL]rx
any-other -
```

```
Ext Attribute:
  owning-user root
  owning-group root
```

Then all aspects of the file system access control and ownership for this file is managed by TAMOS Agent policy.

You can also manage the UNIX special file permissions, **setuid**, **setgid**, and the sticky bit, for the file or directory in policy by using the new extended attribute **special-bits**. Unlike the **owning-user** and **owning-group** extended attributes, the **special-bits** extended attribute may only be applied on the protected object (not on the attached ACL). The value of the **special-bits** extended attribute is a comma-delimited list, with the desired attribute specified strictly in order with no spaces.

In order to set the **setuid** bit, the string **setuid** must be specified, followed by the text string **setgid** (if that is to be set), followed by the text string **sticky** (if the sticky bit is to be set). If the attribute value does not follow this format, or includes additional bits, no special permissions will be applied to the file and a warning will be logged.

If no special permissions should be applied to the file, the attribute should be set to *none*. If the **special-bits** extended attribute is not applied to the File resource protected object, then the file's special permissions will be left untouched.

Table 20 summarizes how the TAMOS Agent maps ACL entries from a Tivoli Access Manager ACL to entries in a POSIX style ACL.

Table 20. Mapping of TAM ACL entries to POSIX ACL entries

Tivoli Access Manager ACL Entry	POSIX ACL entries
owning-user user	Modifies the owning user's permissions to match the specified permissions.
all other user entries	<i>user:user:perms</i>
owning-group group	Modifies the owning group's permissions to match the specified permissions.
all other group entries	<i>group:group:perms</i>
any-other	Modifies the permissions for other to match the specified permissions.
unauthenticated	Not applicable.

Non-existence of resource: If the file or directory represented by the File resource defined in the Tivoli Access Manager for Operating Systems policy does not actually exist on the target system, the policy is ignored and a message is logged.

Furthermore, if at some later stage the file or directory in question is created on the target system, the TAMOS Agent will, after some configurable period of time,

detect that it has been created and apply the associated permissions. For further details on configuring this policy reconciliation period, see “Native Policy Manager” on page 2.

Support for file systems with NFSv4-style ACLs (Solaris ZFS and AIX JFS2 filesystems)

For file systems with NFSv4-style ACLs, such as the Solaris ZFS file system and the AIX JFS2 file system, the TAMOS Agent maps OSSEAL Actions and Tivoli Access Manager Base Actions to UNIX permissions as follows:

Table 21. Mapping OSSEAL actions to UNIX permissions for NFSv4 ACLs

OSSEAL actions		NFSv4 permissions
Always granted		R (read-xattr), a (read attributes), c (read ACL), s (synchronize)
For files:	N (create)	w (write), W (write-xattr)
	R (rename)	w (write), p (append), d (delete)
	U (utime)	A (write-attr)
	o (chown)	o (write-owner)
	p (chmod)	W (write-xattr), C (write-acl)
	r (read)	r (read)
	w (write)	w (write), p (append), W (write-xattr)
	x (execute)	x (execute)
For directories:	D (chdir)	x (search)
	N (create)	w (write), p (append) , W (write-xattr)
	R (rename)	w (write), p (append), d (delete), D (delete-child)
	U (utime)	A (write-attr)
	d (delete)	d (delete), D (delete-child)
	l (list directory)	r (read / list)
	o (chown)	o (write-owner)
	p (chmod)	W (write-xattr), C (write-acl)
Tivoli Access Manager base action:		
For directories:	T (traverse)	x (search)

The W (write-xattr), C (write-acl), and A (write-attr) NFSv4 permissions are implicitly granted to the owning user of the file system resource on both AIX and Solaris. The TAMOS Agent will not attempt to deny the owning user these permissions when doing the mapping, as the resulting NFSv4 ACL would be rejected on AIX.

All other OSSEAL actions and Tivoli Access Manager base actions are ignored.

The set of permissions always granted when setting basic UNIX permissions with **chmod** is the same as that which is granted for POSIX ACLs (described in the previous section). To use a simple illustration, ‘R’, ‘a’, and ‘c’ are required to run **ls** on a directory where you have ‘r’, and they have no effect without other permissions on the file or its containing directory.

The NFSv4 ACL format specifies a more flexible evaluation mechanism than Tivoli Access Manager provides. The TAMOS Agent takes the approach of constructing NFSv4 ACLs that will evaluate the same way as the Tivoli Access Manager ACL would. In the Tivoli Access Manager evaluation model, if there is an ACL entry for a user, then only that entry can grant the user permissions.

To produce this effect in an NFSv4 ACL, an **allow** entry is created, granting the appropriate permissions, followed by a **deny** entry denying all other permissions. In the Tivoli Access Manager evaluation model, any of the group entries that apply to the user can grant any or all of the permissions required to perform an operation. Mapping this to the NFSv4 model, the TAMOS agent first adds **allow** entries for each of the group entries, then **deny** entries for each, denying the permissions not granted by the **allow** entry. Finally, the Tivoli Access Manager any-other ACL entry is mapped to an NFSv4 **everyone@** entry. The **owning-user** entry and **owning-group** group entry are mapped to NFSv4 **owner@** and **group@** entries, but otherwise treated as regular user and group entries.

The following table summarizes how the TAMOS Agent maps ACL entries from a Tivoli Access Manager ACL to entries in a NFSv4 style ACL

Table 22.

Tivoli Access Manager for Operating Systems Entry	NFSv4 entries
'owning-user' user	owner@:<perms>:allow owner@:<~perms>:deny
all other user entries	user:<user>:<perms>:allow user:<~perms>:deny
'owning-group' group	group@:<perms>:allow group@:<~perms>:deny
all other group entries	group:<group>:<perms>:allow group:<group>:<~perms>:deny
any-other	everyone@:<perms>:allow
unauthenticated	not applicable

The **owning-user**, **owning-group**, and **special-bits** attributes will still be applied as they are for POSIX-style ACLs.

For example, consider the following Tivoli Access Manager ACL:

```
group dba T[OSSEAL] rwx
group dbread T[OSSEAL] rx
any-other T
```

When attached to a file system resource representing a file, this Tivoli Access Manager ACL would set entries in the NFSv4 ACL associated with the file for the groups **dba** and **dbread**, as shown.

Solaris

The following output corresponds to that shown by the Solaris **ls -lV** command:

```
-rw-r-----+ 1 root    root          some-file
owner@:rw-p--aARWcCos:-----:allow
owner@:--x-dD-----:-----:deny
group@:r-----a-R-c--s:-----:allow
group@:-wxpdD-A-W-Co-:-----:deny
```

```

group:dba:rxp--a-RWc--s:-----:allow
group:dbread:r-x---a-R-c--s:-----:allow
group:dba:----dD-A---Co:-----:deny
group:dbread:-w-pdD-A-W-Co:-----:deny
everyone@:-----a-R-c--s:-----:allow

```

AIX The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type   NFS4
*
*
* Owner: root
* Group: system
*
s:(OWNER@):   a      rwpRWaAdcCs
s:(OWNER@):   d      xDo
s:(GROUP@):   a      rRadcs
s:(GROUP@):   d      wpWxDaCo
g:dba:  a      rwpRWxacs
g:dbread:    a      rRxacs
g:dba:  d      DAdCo
g:dbread:    d      wpWDAdCo
s:(EVERYONE@): a      Racs

```

When attached to a file system resource representing a directory, with the T (traverse) permission mapped, the NFSv4 ACL associated with the directory would have the following entries (the following output corresponds to that shown by the **ls -lV** command):

Solaris

The following output corresponds to that shown by the Solaris **ls -lV** command:

```

drwxr-x--x+  root  root      some-directory
              owner@:rxp--aARWcCos:-----:allow
              owner@:----dD-----:-----:deny
              group@:r-x---a-R-c--s:-----:allow
              group@:-w-pdD-A-W-Co:-----:deny
              group:dba:--x---a-R-c--s:-----:allow
              group:dbread:--x---a-R-c--s:-----:allow
              group:dba:rw-pdD-A-W-Co:-----:deny
              group:dbread:rw-pdD-A-W-Co:-----:deny
              everyone@:--x---a-R-c--s:-----:allow

```

AIX The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type   NFS4
*
*
* Owner: root
* Group: system
*
s:(OWNER@):   a      rwpRWxDaAdcCs
s:(OWNER@):   d      o
s:(GROUP@):   a      rRxadcs
s:(GROUP@):   d      wpWDaCo
g:dba:  a      Rxacs
g:dbread:    a      Rxacs
g:dba:  d      rwpWDAdCo
g:dbread:    d      rwpWDAdCo
s:(EVERYONE@): a      Racs

```

If the Tivoli Access Manager ACL shown above is now amended to include an **owning-user** and **owning-group** entry as follows:

```

user owning-user T[OSSEAL]rw
group owning-group T[OSSEAL]r
group dba T[OSSEAL]rw
group dbread T[OSSEAL]r
any-other T

```

Then the user and group entries of the file's NFSv4 ACL are now managed by the TAMOS Agent as shown.

Solaris

The following output corresponds to that shown by the Solaris **ls -lV** command:

```

-rw-r-----+ 1 root    root      some-file
                owner@:rw-p--a-RWc--s:-----:allow
                owner@:--x-dD-----o:-----:deny
                group@:r-----a-R-c-s:-----:allow
                group@:-wxpdD-A-W-Co:-----:deny
                group:dba:rw-p--a-RWc--s:-----:allow
                group:dbread:r-----a-R-c-s:-----:allow
                group:dba:--x-dD-A---Co:-----:deny
                group:dbread:-wxpdD-A-W-Co:-----:deny
                everyone@:-----a-R-c-s:-----:allow

```

AIX The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type   NFS4
*
*
* Owner: root
* Group: system
*
s:(OWNER@):   a      rwpRWacs
s:(OWNER@):   d      xDdo
s:(GROUP@):   a      rRacs
s:(GROUP@):   d      wpWxDAdCo
g:dba: a      rwpRWacs
g:dbread:     a      rRacs
g:dba: d      xDAdCo
g:dbread:     d      wpWxDAdCo
s:(EVERYONE@): a      Racs

```

The **owning-user**, **owning-group**, and **special-bits** attributes will still be applied in exactly the same way as shown in the POSIX ACL example and therefore are not shown here.

Support for file systems with AIX Classic style ACLs (AIX JFS and JFS2 filesystems)

For file systems with AIX Classic ACLs, such as the JFS file system, the TAMOS Agent maps OSSEAL actions and Tivoli Access Manager base actions to UNIX file system permissions as follows:

Table 23. Mapping OSSEAL actions to UNIX permissions

OSSEAL actions		UNIX permissions
For files:	r (read)	r
	w (write)	w
	x (execute)	s

Table 23. Mapping OSSEAL actions to UNIX permissions (continued)

OSSEAL actions		UNIX permissions
For directories:	l (list directory)	r
	N (create), o (chown), p (chmod), R (rename), d (delete), U (utime)	w
	D (chdir)	x
Tivoli Access Manager base action:		
For directories:	T (traverse)	x

All other OSSEAL actions and Tivoli Access Manager base actions are ignored.

The AIX Classic ACL format allows for a broader specification than Tivoli Access Manager provides with respect to when an ACL entry should be used. The TAMOS Agent takes the approach of constructing AIX Classic ACLs that will evaluate in the same way as the Tivoli Access Manager ACL. In the Tivoli Access Manager evaluation model, if there is an ACL entry for a user, then only that entry can grant the user permissions.

To produce this effect in an AIX Classic ACL, only the specify ACL entry type is used. The **permit** and **deny** ACL entry types are not used by the TAMOS Agent. The specify ACL entry contains only the permissions that the user or group is allowed, as described in the ACL. Furthermore, each AIX Classic ACL entry will only refer to a single user or group. The ability to specify a number of groups in a single AIX Classic ACL entry and have the entry used only if the user is a member of all the groups has no equivalent in Tivoli Access Manager ACLs, so is not used.

Table 24. AIX Classic ACL entries

Tivoli Access Manager for Operating Systems Entry	AIX Classic entries
owning-user user	owner(<owning-user-name>): <perms>
owning-group group	group(<owning-group-name>): <perms>
any-other	others: <perms>
all other user entries	specify: <perms> u:<user>
all other group entries	specify: <perms> g:<group>
unauthenticated	not applicable

The **owning-user**, **owning-group**, and **special-bits** attributes will still be applied, as they are for POSIX-style ACLs.

As an example, consider the following Tivoli Access Manager ACL:

```
group dba T[OSSEAL]rwx
group dbread T[OSSEAL]rx
any-other T
```

When attached to a file system resource representing a file, this Tivoli Access Manager ACL would set entries in the AIX Classic ACL associated with the file for the groups **dba** and **dbread**, as shown.

The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type  AIXC
*
attributes:
base permissions
    owner(root):  rw-
    group(system): r--
    others:  ---
extended permissions
    enabled
    specify  rw-      g:dba
    specify  r-x      g:dbread

```

When attached to a file system resource representing a directory, with the **T** (traverse) permission mapped, the AIX Classic ACL associated with the directory would have the following entries.

The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type  AIXC
*
attributes:
base permissions
    owner(root):  rw-
    group(system): r-x
    others:  ---
extended permissions
    enabled
    specify  --x      g:dba
    specify  --x      g:dbread

```

If the Tivoli Access Manager ACL shown above is now amended to include an **owning-user** and **owning-group** entry as follows:

```

user owning-user T[OSSEAL]rw
group owning-group T[OSSEAL]r
group dba T[OSSEAL]rw
group dbread T[OSSEAL]r
any-other T

```

Then the user and group entries of the file's AIX Classic ACL are now managed by the TAMOS Agent as shown.

The following output corresponds to that shown by the AIX **aclget** command:

```

*
* ACL_type  AIXC
*
attributes:
base permissions
    owner(root):  rw-
    group(system): r--
    others:  ---
extended permissions
    enabled
    specify  rw-      g:dba
    specify  r--      g:dbread

```

The **owning-user**, **owning-group**, and **special-bits** attributes will still be applied in exactly the same way as shown in the POSIX ACL example and therefore are not shown here.

Audit policy

Tivoli Access Manager for Operating Systems provides two ways to define policy for auditing, resource-level audit policy and per-user level audit policy. These are described in the following sections.

For more information about TAMOS Agent auditing support, see “TAMOS Agent Auditing” on page 64.

Resource-level audit policy

TAMOS Agent supports the Tivoli Access Manager for Operating Systems resource-level audit policy, which provides the ability to enable auditing for specified resources. Resource-level auditing is defined by:

1. using Tivoli Access Manager protected object policy (POPs) access controls to enable auditing, and then
2. attaching the POP to the protected resource in the Tivoli Access Manager object space.

Defined resource-level audit policy enables resource-level auditing in exactly the same way as the Tivoli Access Manager for Operating Systems product for any authorization decisions made by the TAMOS Agent (for example, for Sudo policy or login location policy access decisions).

The TAMOS Agent also supports resource-level audit policy for file resources, by mapping the defined Tivoli Access Manager for Operating Systems policy to native operating system audit settings. Mapping from Tivoli Access Manager for Operating Systems policy to native operating system audit settings will vary by platform, according to the capabilities of native audit subsystem.

Note: Resource-level audit policy is not supported for file resources specified using wildcard characters.

Resource-level audit policy mapping to native operating system settings

File audit settings are defined by POPs attached to policy objects, defined under the File resource of a Tivoli Access Manager for Operating Systems policy branch. The supported POP audit levels are **permit** and **deny**. These can be qualified by extended attributes, which reduce the set of Tivoli Access Manager for Operating Systems actions that should trigger the generation of an audit record.

The precise mapping of Tivoli Access Manager for Operating Systems actions to target system audit policies is dependent on the operating system running at the target system. The operating system-specific mappings are defined in the following sections.

Solaris: Solaris auditing defines a number of audit event types in the `/etc/security/audit_event` file. This file identifies the events that may be generated and the audit classes that include each event. Audit classes define categories of related events, and are described in the `/etc/security/audit_classes` file.

The TAMOS Agent uses Solaris audit event classes when mapping from Tivoli Access Manager for Operating Systems actions to Solaris audit configuration. Therefore, the Solaris audit classes should not be renamed or changed substantially without understanding how such action might impact the mapping of Tivoli Access Manager for Operating Systems policy to Solaris audit classes.

The tables below define how Tivoli Access Manager for Operating Systems file audit policy maps to Solaris audit settings.

Table 25. file audit policy mapping to Solaris settings

POP audit level value	Map to these Solaris Audit classes
permit	the complete set of audit classes listed in the table below, prepended by the + (plus) sign.
deny	the complete set of audit classes listed in the table below, prepended by the - (minus) sign.

Note: These levels are not mapped to **+all**, **-all**, or **all**, because of the large volume of audit records which this could generate. The mapping is restricted to only enable the generation of those audit events corresponding to Tivoli Access Manager for Operating Systems permissions.

For file audit policy POPs that do have the extended attributes **audit_permit_actions** and **audit_deny_actions** defined (which limit the auditing being enabled to just the actions specified), the mapping of Tivoli Access Manager for Operating Systems actions to native operating system audit classes (Solaris) is as follows:

Table 26. audit class mapping on Solaris

Tivoli Access Manager for Operating Systems action	Maps to Solaris audit class
read (r)	fr, fa
write (w)	fw
create (N)	fc
execute (ex)	ex
chown (o)	fm
chmod (p)	fm
chdir (D)	pm
rename (R)	fc or fd
delete (d)	fd
utime (u)	fm
list (l)	fr

Note: Solaris 10 already optimizes out audit records pertaining to read-only operations for "public objects" (where a public object is one owned by root and world readable). If you want read access to such objects audited, you can enable this by running the command:

```
auditconfig -setpolicy +public
```

AIX: Tivoli Access Manager Protected Object Policies (POPs) are used to define per-resource based auditing which can then be attached to File resources defined in Tivoli Access Manager for Operating Systems policy. For each File resource with an attached POP with auditing enabled, the TAMOS Agent will create an entry in the AIX system file `/etc/security/audit/objects` with a stanza name equal to the path name of the file. The file attributes defined for each stanza have the following format:

`access_mode = "audit_event"`

The **access_mode** value will depend on the defined TAMOS policy. The mappings from TAMOS policy is shown in Table 27 and Table 28.

The AIX auditing configuration does not distinguish between auditing of permitted and denied operations, so if the POP contains both **permit** and **deny** audit levels, the **access_mode** values set will be the union of those set by the **permit** level and those set by the **deny** level.

The default value for **audit_event** is `FILE_Open`. The information audited by the `FILE_Open` audit event includes the file name, the user, the flags and mode use to open the file. The **audit_event** used for a particular access mode can be configured by setting an extended attribute on the POP, as shown in Table 29 on page 51.

Table 27 shows the mapping for the POP audit level value to AIX file attributes in the `/etc/security/audit/objects` file.

Table 27. mapping POP audit levels to AIX file attributes

POP audit level value	map to these AIX access modes = "audit_event"
permit	r = "FILE_Open" w = "FILE_Open" x = "FILE_Open"
deny	r = "FILE_Open" w = "FILE_Open" x = "FILE_Open"

For POPs that do have the extended attributes **audit_permit_actions** and **audit_deny_actions** defined (which limit the auditing being enabled to just the actions specified) mapping the Tivoli Access Manager for Operating Systems actions to AIX file attributes in the each stanza is performed as shown in Table 28:

Table 28.

Tivoli Access Manager for Operating Systems action	map to this AIX audit file attribute
read (r)	r= "FILE_Open"
write (w)	w= "FILE_Open"
create (N)	w= "FILE_Open"
execute (x)	x= "FILE_Open"
chown (o)	w= "FILE_Open"
chmod (p)	w= "FILE_Open"
chdir (D)	x= "FILE_Open"
rename (R)	w= "FILE_Open"
delete (d)	w= "FILE_Open"
utime (U)	w= "FILE_Open"
list (l)	r= "FILE_Open", x= "FILE_Open"
kill (K)	not supported

Table 29 shows the mapping between the new POP extended attributes and the access modes they represent. The value of the extended attribute is the **audit_class** that is used for the specified access mode.

Table 29.

Protected Object Policy extended attribute	AIX audit file access_mode
aix_audit_read_event	r
aix_audit_write_event	w
aix_audit_execute_event	x

After the TAMOS Agent modifies the AIX `/etc/security/audit/objects` file it stops and restarts the auditing system on AIX using the AIX audit command.

Existing entries in `/etc/security/audit/objects` unrelated to files explicitly defined in the Tivoli Access Manager for Operating Systems policy will be left unchanged. The TAMOS Agent managed entries will be put after any existing entries.

As an example of the mapping between policy and AIX configuration file entries, consider a POP created as follows:

```
pop create testpop
pop modify testpop set audit-level permit,deny
pop modify testpop set attribute audit_permit_actions w
pop modify testpop set attribute audit_deny_actions r
pop modify testpop set attribute aix_audit_write_event S_PASSWD_WRITE
pop modify testpop set attribute aix_audit_read_event S_PASSWD_READ
```

This POP specifies that failed read operations and successful write operations should be audited when attached to a File resource. However, because the AIX Auditing system cannot be configured to audit only successful or unsuccessful actions, all read and write attempts will be audited. The **aix_audit_read_event** and **aix_audit_write_event** attributes control which audit class is used for the read and write operations, respectively. If this POP was attached to the File resource object:

`/OSSEAL/policy-branch/File/etc/security/passwd`

then the `/etc/security/audit/objects` file would be updated to include the following stanza managed by the TAMOS Agent:

```
/etc/security/passwd:
r = "S_PASSWD_READ"
w = "S_PASSWD_WRITE"
```

If the POP was then modified to remove the **audit_permit_actions** attribute, the stanza would be updated to the following:

```
/etc/security/passwd:
r = "S_PASSWD_READ"
w = "S_PASSWD_WRITE"
x = "FILE_Open"
```

The removal of the **audit_permit_actions** attribute means that the permit level will now audit **r**, **w** and **x** actions. Since the **aix_audit_execute_event** attribute is not set, the default value of `FILE_Open` is used.

The AIX system file `/etc/security/audit/events` contains information about the audit classes in AIX.

Note: Depending on what is set in the `/etc/security/audit/streamcmds` file, restarting the audit system will clean up the stream auditing file, but the **binaudit** devices will always remain. While the auditing has stopped, no events will be audited. The default entry in the `/etc/security/audit/streamcmds` file is:

```
/usr/sbin/auditstream | auditpr > /audit/stream.out &
```

This overwrites **stream.out** every time it is invoked (auditing restarted).
Modifying this to:

```
/usr/sbin/auditstream | auditpr >> /audit/stream.out &
```

will mean that restarting the audit system does not erase audit logs. However, the AIX Security Expert pages recommend that **stream** auditing not be used, and that **bin** mode should be used instead. To turn on auditing on both **bin** mode and **stream** mode, the `/etc/security/audit/config` file should be modified to show:

```
start:
    binmode = on
    streammode = on
```

The AIX Information Center contains the AIX Security Expert Audit Policy Recommendations at: http://publib.boulder.ibm.com/infocenter/systems/topic/com.ibm.aix.security/doc/security/aix_sec_expert_aud_policy_settings.htm?tocNode=int_3263

User-level audit policy

TAMOS Agent supports the Tivoli Access Manager for Operating Systems user-level audit authorization and audit trace policy, which provides the ability to enable auditing for specified users by mapping this policy to native operating system audit settings. The precise mapping of Tivoli Access Manager for Operating Systems audit levels to target system audit policies is dependent on the operating system running at the target system. The operating system-specific mappings are defined in the following sections. Defined user-level audit authorization policy does not enable any auditing by the TAMOS Agent itself. It is not supported as a way to enable auditing for authorization decisions made by the TAMOS Agent (for example, for Sudo policy or login location policy access decisions).

User audit levels can be set by defining resources that represent the audit levels.

User-level audit authorization resources are represented by defining an object name with a resource type of **AuditAuth** as shown:

- `/OSSEAL/policy-branch/AuditAuth/User` or
- `/OSSEAL/policy-branch/AuditAuth/Group`

AuditAuth resource policy supports the following audit levels:

permit

Enables the generation of audit records for all authorization decisions that permit access to protected resources for the user.

deny

Enables the generation of audit records for all authorization decisions that deny access to protected resources for the user.

loginpermit

Enables the generation of audit records for all login-related authorization decisions that permit a login by the user.

logindeny

Enables the generation of audit records for all login-related authorization decisions that deny a login by the user.

all Turns on all audit levels.

none A special case. Indicates that no audit records should be generated for the user.

To set the **AuditAuth** levels, objects of the following format are created:

```
/OSSEAL/policy-branch/AuditAuth/User/user-name/audit-level
/OSSEAL/policy-branch/AuditAuth/Group/group-name/audit-level
```

Note: The user or group names specified should be the UNIX user or group names. This user may also be defined in the Tivoli Access Manager user registry; but there is no requirement that the specified user or group name be a Tivoli Access Manager user.

AuditTrace resources can be defined for a specific UNIX user by using the **User** keyword in the resource definition. User-level trace resources are represented by defining an object name with a resource type of **AuditTrace** as shown:

```
/OSSEAL/policy-branch/AuditTrace/User
```

Supported user-level trace audit levels are:

exec Track program invocations initiated by the **exec()** system call that occur in processes that descend from a login event for the specified user.

exec_1 This level is treated exactly the same as **exec** by the TAMOS Agent product.

file Tracks all accesses to protected file resources by the specified user.

all Activates both **exec** and **file** audit levels.

none A special case that indicates that no audit records should be generated for the specified user.

To set the **AuditTrace** levels, objects of the following format are created:

```
/OSSEAL/policy-branch/AuditTrace/User/user-name/trace-level
```

Note: The user name specified should be the UNIX user name. This user may also be defined in the Tivoli Access Manager user registry; but there is no requirement that the specified user name be a Tivoli Access Manager user.

The TAMOS Agent supports user-level audit policy by mapping the defined Tivoli Access Manager for Operating Systems policy to native operating system audit settings. Mapping from Tivoli Access Manager for Operating Systems policy to native operating system audit settings will vary by platform, according to the capabilities of native audit subsystem.

Per-user audit policy mapping to native operating system settings

Solaris:

Per-user AuditAuth and AuditTrace audit policy mapping

Per-user authorization audit settings are defined using either of the following Tivoli Access Manager for Operating Systems object space policies:

- /OSSEAL/policy-branch/AuditAuth/User or
- /OSSEAL/policy-branch/AuditAuth/Group

The **per-user** audit policy will be mapped to Solaris **per-user** audit settings that are defined in the file /etc/security/audit_user.

Per-user authorization audit levels map to Solaris audit classes according to the following table:

Table 30. mapping per-user authorization audit levels to Solaris

loginpermit	+lo
logindeny	-lo
permit	+ ex, fa, fc, fd, fr, fm, fw, lo and pm
deny	- ex, fa, fc, fd, fr, fm, fw, lo and pm
all	ex, fa, fc, fd, fr, fm, fw, lo and pm
none	auditing is turned off for this user.

Since Solaris audit configuration does not include the capability to specify auditing on a group basis, per-group audit policy specified in Tivoli Access Manager for Operating Systems policy is mapped to **per-user** audit setting in the /etc/security/audit_user file for each member of the group. Group membership is resolved at the time policy is applied and also during each policy reconciliation.

Activity trace auditing can be specified by defining policy under the **AuditTrace** resource of a Tivoli Access Manager for Operating Systems policy branch as follows.

/OSSEAL/policy-branch/AuthTrace/User

For **AuditTrace** policy, the mapping is as follows:

Table 31. mapping AuditTrace audit levels to Solaris

AuditTrace audit level value	Map to these Solaris Audit classes
file	ex, fa, fc, fd, fr, fm, and fw
all	ex, fa, fc, fd, fr, fm, and fw
exec	ex
exec_l	ex
none	create an entry in audit_user that turns off all auditing for the classes list in the table above, this will override the default system auditing for the user.

Audit policy verification/reconciliation

For the native Solaris audit classes controlled by the TAMOS Agent, the policy specifies exactly which audit classes must be enabled.

For example, if a configured Tivoli Access Manager for Operating Systems audit policy maps to **-fc** (audit denied file creations), and the currently configured Solaris audit policy (for example in the **audit_user** file) is **fc** (audit permitted and denied file creations), that is considered a difference as compared to our policy. In this situation the TAMOS Agent would therefore change the configured Solaris audit setting back to **-fc** as part of its policy reconciliation processing.

Audit classes that our policy does not control (such as **nt**, **ip**, **ss**, etc.), are not managed by the TAMOS Agent product. Their presence or absence from the Solaris audit configuration files is ignored.

Note: Audit policy changes will not take effect for a given user until that user next logs into the target Solaris operating system.

AIX: **AuditAuth** and **AuditTrace** policy is used to define per-user based auditing. In order to facilitate mapping Tivoli Access Manager for Operating Systems **AuditAuth** and **AuditTrace** policy to AIX audit events, the TAMOS Agent defines new classes which are mapped to AIX audit events. These new classes are added to the classes stanza in the AIX `/etc/security/audit/config` file.

AIX limits the number of audit classes that can be specified in the `/etc/security/audit/config` file to 31. So if there are more than 27 audit classes already defined, the TAMOS Agent is unable to create its audit classes and cannot map **AuditAuth** and **AuditTrace** policy to native policy.

Table 32 shows the mapping between the TAMOS Agent defined classes and the AIX audit events:

Table 32. mapping audit classes to AIX audit events

TAMOS Agent audit class	AIX audit events
TAMOS_exec	PROC_Execute, PROC_LPExecute
TAMOS_authzn	FS_Chdir, USER_SU, PROC_SetUserIDs, PROC_Kill, USER_Login, FILE_Open, FILE_Link, FS_Mkdir, FILE_Symlink, FILE_Rename, FILE_Utimes, FILE_Truncate, FS_Rmdir, FILE_Owner, FILE_Fchown, FILE_Mode, FILE_Fchmod, FILE_Read, FILE_ReadXacl, FILE_FReadXacl, FILE_Write, FILE_WriteXacl, FILE_FWriteXacl, PROC_Execute, PROC_LPExecute
TAMOS_file	FILE_Open, FILE_Link, FS_Mkdir, FILE_Symlink, FILE_Rename, FILE_Utimes, FILE_Truncate, FS_Rmdir, FILE_Owner, FILE_Fchown, FILE_Mode, FILE_Fchmod, FILE_Read, FILE_ReadXacl, FILE_FReadXacl, FILE_Write, FILE_WriteXacl, FILE_FWriteXacl
TAMOS_login	USER_Login

Using the new TAMOS Agent defined audit classes, for **AuditAuth** policy the mapping will be as shown in Table 33:

Table 33. mapping AuditAuth audit levels to AIX classes

AuditAuth audit level value	map to these Audit classes
all	TAMOS_authzn, TAMOS_login
permit	TAMOS_authzn
deny	TAMOS_authzn
loginpermit	TAMOS_login
logindeny	TAMOS_login
none	No entry for the user

Using the new TAMOS Agent defined audit classes, for **AuditTrace** policy, the mapping will be as shown in Table 34:

Table 34. mapping AuditTrace audit levels to AIX classes

AuditTrace audit level value	map to these Audit classes
file	TAMOS_file
all	TAMOS_file, TAMOS_exec
exec	TAMOS_exec
exec_l	TAMOS_exec
none	No entry for the user

The first time the TAMOS Agent is required to map **AuditAuth** or **AuditTrace** policy it must create the audit classes shown in Table 31. On subsequent policy updates the TAMOS Agent will verify that these audit classes are still defined and still contain the audit events shown in Table 32 on page 55. If an update to the audit classes is required, the TAMOS Agent must restart the AIX audit system to affect these changes. For more information about the effect of restarting the audit system, refer to 52.

To manage the application of **AuditAuth** and **AuditTrace** policies, the TAMOS Agent uses the AIX **chuser** command to change the per-user audit. The **chuser** command does not require the audit system to be restarted for the changes to take effect.

Note: The TAMOS Agent native policy manager assumes it is managing the TAMOS Agent defined audit classes (**TAMOS_exec**, **TAMOS_authzn**, **TAMOS_file** and **TAMOS_login**) for all users. Other audit classes are not managed by the TAMOS Agent and if users have these assigned to them, they will be retained after policy application and reconciliation. However, if the TAMOS Agent-specific audit classes are assigned to users by means other than through Tivoli Access Manager policy, these will not be retained after policy application or reconciliation.

Chapter 4. Comparisons with Tivoli Access Manager for Operating Systems 6.0

This chapter describes important differences between Tivoli Access Manager for Operating Systems version 6.0 and the TAMOS Agent. This information is included to help readers who are familiar with the existing Tivoli Access Manager for Operating Systems product, and who want to better understand differences with the TAMOS Agent.

Policy support

The TAMOS Agent runs entirely as a user-level endpoint. Without the Tivoli Access Manager for Operating Systems kernel extensions and the capability to intercept system calls, it is no longer possible to support the complete set of authorization policy supported by the existing Tivoli Access Manager for Operating Systems version 6.0 product.

The TAMOS Agent employs a hybrid approach to supporting the existing Tivoli Access Manager for Operating Systems authorization policy. The TAMOS Agent makes authorization decisions for policy types such as Sudo, Login location, and Login holiday policy. Access controls related to file system resources, login activity policy, password management and audit policy (POPs attached to File resources, user-level auditing policy) will be mapped to native operating system security and audit settings. The subsets of policies mapped are dictated by the ability of the operating system to represent the policy natively. Authorization policies related to network services and surrogate operations are not supported by the TAMOS Agent.

The following tables summarize how Tivoli Access Manager for Operating Systems policy is supported by the TAMOS Agent.

- Table 35 on page 58 shows the policy that is supported by TAMOS Agent making an authorization decision, similar to the current TAMOS product.
- Table 36 on page 59 shows the policy that is supported by TAMOS Agent mapping the policy to native operating system settings.
- Table 37 on page 60 shows the policy which the TAMOS Agent does not support.

A TAMOS Agent endpoint should not be configured to use a policy branch that was created during the configuration of a Tivoli Access Manager for Operating Systems product endpoint since the default policy is not applicable to a TAMOS Agent endpoint. Also, a Tivoli Access Manager for Operating Systems product should not be configured to use a policy branch created during the configuration of a TAMOS Agent endpoint since the necessary default policy has not been defined in that policy branch. Policy branches created explicitly as part of a multi-branch configuration can however be shared.

Table 35. Policy supported by TAMOS Agent making an authorization decision

Policy / Feature	Notes
Sudo policy	<p>The system audit ID is used to identify the invoking user rather than TAMOS accessor ID Global and resource-level audit policy applies to Sudo authorization decisions.</p> <p>AuditAuth audit policy does not apply to Sudo authorization decisions.</p> <p>Wildcard support for defining policy specific to certain command arguments is as documented in the <i>Tivoli Access Manager for Operating Systems Administration Guide</i>.</p>
Login location policy, time-of-day restrictions, holiday policy	<p>Requires PAM to be enabled on the system. A new TAMOS Agent PAM module is provided with different options specific to the TAMOS Agent.</p> <p>Remote login location policy is enforced based on the hostname supplied by PAM to the TAMOS Agent PAM module. By default, non-fully qualified host names will be mapped only to default remote terminal policy. This is a security consideration. Refer to the PAM module documentation above for further details</p> <p>Global and resource-level audit applies to Login location, time-of-day restrictions, and holiday authorization decisions.</p> <p>AuditAuth audit policy does not apply to Login location, time-of-day restrictions, and holiday authorization decisions.</p> <p>Wildcard support is as documented in the <i>Tivoli Access Manager for Operating Systems Administration Guide</i>.</p> <p>Access restrictions are not supported.</p>

Table 36. Policy supported by TAMOS Agent mapping the policy to native operating system settings

Policy	Notes on TAMOS Agent policy mapping
File policy	<p>File system resource policy is specified under the /OSSEAL/policybranch/File object space and is supported by mapping the policy to native file system access control settings.</p> <p>ACL inheritance is not supported.</p> <p>The mapping of Tivoli Access Manager for Operating Systems permission bits to system permissions depends on type of file system containing the target file resource.</p> <p>Access restrictions are not supported.</p> <p>Wildcards may only be specified on the leaf of a file specification in TAMOS Agent policy and are not applied recursively to subdirectories.</p> <p>Tivoli Access Manager for Operating Systems kill permission ([OSSEAL]K) is not supported.</p> <p>Resource-level auditing (POPs attached to File resources) are mapped to native operating system audit settings. The effect of this varies by platform;</p> <ul style="list-style-type: none"> • Solaris does not have per-file audit settings. • AIX has per-file audit settings; POPs attached on wildcard resources not supported. <p>Global auditing does not apply to file resources.</p>
Login activity policy	<p>Login activity policy is specified with the Tivoli Access Manager for Operating Systems Login activity extended attributes. The policy is supported by mapping the policy to native operating system settings. Only a subset of the login activity attributes will take effect, and that subset is platform specific.</p> <p>User exceptions override the defined default login activity policy on a per attribute basis, rather than entirely replacing the default policy.</p>
Password Management policy	<p>Password management policy is specified with the Tivoli Access Manager for Operating Systems Password management extended attributes. The policy is supported by mapping the policy to native operating system settings.</p> <p>Only a subset of the password policy attributes is supported, and that subset is platform specific.</p> <p>User exceptions override the defined default login activity policy on a per attribute basis, rather than entirely replacing the default policy.</p>

Table 36. Policy supported by TAMOS Agent mapping the policy to native operating system settings (continued)

Policy	Notes on TAMOS Agent policy mapping
User-level audit and trace policy	User-level audit and trace policies are specified under the /OSSEAL/policybranch/AuditAuth and /OSSEAL/policybranch/AuditTrace object space. They are supported by mapping the policies to native operating system settings.

Table 37. Tivoli Access Manager for Operating Systems policies not supported by the TAMOS Agent

Policy	Notes on policies not supported by the TAMOS Agent
Incoming network connection (NetIncoming) policy	Specified under the /OSSEAL/policybranch/NetIncoming object space. This policy type is not supported.
Outgoing network connection (NetOutgoing) policy	Specified under the /OSSEAL/policybranch/NetOutgoing object space. This policy type is not supported.
User and group identity change (Surrogate) policy	Specified under the /OSSEAL/policybranch/Surrogate object space. This policy type is not supported.
Trusted Computer Base (TCB) policy	Specified under the /OSSEAL/policybranch/TCB object space. This policy type is not supported.
Access-Restrictions extended attributes	Specified with the Access-Restrictions extended attribute. This policy type is not supported.

Runtime

The TAMOS Agent runs entirely as a user-level endpoint. This section briefly describes the daemon processes and utilities included with the TAMOS Agent. It also describes those Tivoli Access Manager for Operating System daemon processes and utilities not supported and not included with the TAMOS Agent.

Daemon processes

The daemons responsible for the major functions of TAMOS Agent are similar to the Tivoli Access Manager for Operating Systems product daemons:

pdosd The policy and authorization daemon makes authorization decisions for Sudo, Login location, time-of-day restrictions, and holiday policy; it parses File, Login activity, Password management, resource-level audit for File, and user-level audit (**AuditAuth** and **AuditTrace**) policy and creates the Resolved Policy Database (rpdb) xml file.

pdosauditd

The audit daemon receives audit events from other components of TAMOS Agent and manages the audit trail. Only audit events generated by TAMOS Agent (health events generated by the various daemons and authorization events for sudo, login location and login holiday

authorization decisions made by TAMOS Agent) are handled by **pdosauditd**. All other audit events are handled by the host operating systems' own audit facilities.

pdoswdd

The watchdog daemon ensures that the other daemons remain available. The other daemons also monitor each other.

pdoslrd

The Log Router Daemon Makes audit records available for transfer to multiple locations.

pdosnpd

The Native Policy Manager daemon is a new daemon that runs as a Java process and implements the mapping from TAMOS Agent policy to native security settings, using the Native Policy plug-ins to apply the policy to the appropriate operating system files.

The Tivoli Access Manager for Operating Systems 6.0 daemons are not included with TAMOS Agent:

pdoslpmd

The Login Policy and Password Management daemon is no longer supported. The TAMOS Agent maps Login activity and Password management policy to native settings for enforcement by the native operating system.

pdostecd

The Tivoli Enterprise Console Daemon is no longer supported. Audit events produced by TAMOS Agent will not be made available to the Tivoli Enterprise Console.

PAM Module

The TAMOS Agent does includes a new Pluggable Authentication Module (PAM) module, which is responsible for enforcing Login location, Login holiday and Login Time-of-Day restrictions

Utilities

The utilities included with TAMOS Agent are similar to the Tivoli Access Manager for Operating Systems utilities:

pdosaudview

Parses the binary audit log produced by TAMOS Agent into a number of readable formats. Syntax and functionality is the same. For the **-g** resource types parameter, the following resource types are no longer relevant (since the TAMOS Agent will not be generating audit records of this type):

- file,
- netincoming,
- netoutgoing,
- password,
- surrogate,
- tcb,
- trace_exec, and
- trace_file.

For the `-r` reason parameter, the following value is no longer relevant since the TAMOS Agent will not be generating audit records for this reason:

- `user_audit`.

pdosbkup

Backs up TAMOS Agent databases and configuration files. Syntax and functionality is the same.

pdosbranchcfg

Controls the configuration of a TAMOS Agent machine to multiple policy branches. Syntax and functionality is the same.

pdoscfg

Configures the TAMOS Agent. Syntax has changed. See Appendix A, "Command reference," on page 71.

pdoscollview

Views records in a collection file created by **pdoslrld**, the log router daemon. Syntax and functionality is the same.

pdosctl

Sends control messages to selected TAMOS Agent daemons. Syntax and functionality is the same. Support has been added for controlling the **pdosnpsd** daemon. The **pdoslpmd** daemon is no longer included with the TAMOS Agent, hence it is not a valid daemon name to specify.

pdoshla

Manages host name lookaside database. Syntax and functionality is the same.

pdoslradm

Controls the log router daemon and channels in the log router configuration file. Syntax and functionality is the same.

pdosrefresh

Refreshes the Tivoli Access Manager for Operating Systems credentials for the invoking user, specified users, or all currently cached users. Syntax and functionality is the same.

pdosrgyimp

Imports UNIX users and groups into the Tivoli Access Manager user registry. Syntax and functionality is the same.

pdosrstr

Restores Tivoli Access Manager for Operating Systems database and configuration files from a backup file. Syntax and functionality is the same.

pdossudo

Invokes a command as a different UNIX user. Syntax and functionality is the same with one difference in the area of how the command identifies the user executing a Sudo command. For more information, see Appendix A, "Command reference," on page 71.

pdosucfg

Unconfigures the TAMOS Agent. Syntax is the same. Functionality is the same except unconfiguration steps related to the kernel modules have been removed.

pdosversion

Displays the version information for TAMOS Agent. Syntax and functionality is the same.

pdoswhoami

Displays information about the Tivoli Access Manager user. Syntax and functionality is the same.

policyview

Extract policy from the Tivoli Access Manager policy database to a plain text file. Syntax and functionality is the same.

rc.osseal

Starts and stops the TAMOS Agent daemon processes. Syntax and functionality is the same except that steps related to the kernel modules have been removed.

Tivoli Access Manager for Operating Systems 6.0 utilities not included with TAMOS Agent

pdosdestroy

Destroys the Tivoli Access Manager for Operating Systems credentials for the specified users.

pdosent

Creates an entitlement report for a specified user or group.

pdosexempt

Disables Tivoli Access Manager for Operating Systems authorization decisions.

pdoslpinf

Displays information about a user password expiration and grace logins.

pdoslpadm

Perform administrative commands pertaining to the Tivoli Access Manager for Operating Systems Login Activity Database.

pdosobjsig

Manages the Tivoli Access Manager for Operating Systems object signature database.

pdosrespolicy

Displays the protected object name and applied policy for a specified File, NetIncoming, or NetOutgoing resource.

pdosrevoke

Revokes an exemption from Tivoli Access Manager for Operating Systems authorization decisions.

pdosshowuser

Shows various attributes of a specified user.

pdosteccfg

Configures pdostecd, the Tivoli Enterprise Console® daemon.

pdostecucfg

Unconfigures pdostecd, the Tivoli Enterprise Console daemon.

pdosuidprog

Identifies setuid or setgid programs on a system so that an administrator can decide whether they should be included in the Trusted Computing Base (TCB).

pdosunauth

Spawns a shell that executes commands in an unauthenticated environment.

pdoswhois

Displays Tivoli Access Manager for Operating Systems accessor ID information associated with the specified process IDs.

Tivoli Access Manager for Operating Systems 6.0 Kernel extensions and related utilities not included

The existing Tivoli Access Manager for Operating Systems product includes system kernel extensions (collectively known as KOSSEAL) which actively provide enforcement by intercepting the native system calls to make authorization decisions for protected system resources. With the architectural shift in this release, moving the responsibility of enforcement away from Tivoli Access Manager for Operating Systems and into the operating systems themselves, the Tivoli Access Manager for Operating Systems kernel extensions are no longer required. The following KOSSEAL related utilities are also not included with TAMOS Agent:

- kazntrace,
- kossctl,
- kossdump.sh,
- kosserrs,
- kossinfo, and
- rc.kosseal.

TAMOS Agent Auditing

Only audit events generated by TAMOS Agent are handled by **pdosauditd**. All other audit events related to the mapping of TAMOS audit policy to native system settings are handled by the host operating systems' own audit facilities.

The TAMOS Agent will generate audit records for administrative, informational, and health events related to the TAMOS Agent daemons, based on the global audit level settings (in the same way the Tivoli Access Manager for Operating Systems product does).

The TAMOS Agent will generate audit records for authorization decisions based on the global audit level and resource-level auditing settings. The TAMOS Agent will only generate authorization audit records for the Sudo and Login resource type, since this is the only policy against which the TAMOS Agent makes an authorization decision. Note that a login audit record will not be generated if the user login is denied by the operating system authentication mechanism.

TAMOS Agent-generated audit records are in the same format as audit records generated by the Tivoli Access Manager for Operating Systems product. The audit log files can be processed and viewed using the **pdosauditview** utility, or processed by the **pdoslrd** daemon.

Enabling auditing for the TAMOS Agent is similar to enabling Tivoli Access Manager for Operating Systems product auditing.

Global auditing can be enabled using the **pdoscfg** or **pdosctl** commands, exactly as is done with the Tivoli Access Manager for Operating System product. The TAMOS Agent supports a limited set of global audit levels. The following audit levels are supported:

- **admin**,
- **all**,

- **deny**,
- **logindeny**,
- **info**,
- **none**,
- **permit**, and
- **verbose**.

The global **permit** and **deny** audit levels can be tuned for more granular auditing of authorization access decisions, based on the action being performed against the protected resource. The TAMOS Agent only supports specifying the execute (x) and login (L) action bits.

The TAMOS Agent supports the generation of audit records that pertain to its health, in the same way as the Tivoli Access Manager for Operating Systems product. This is enabled using the **pdoscfg** command.

Resource-level auditing can be enabled by defining POPs with auditing enabled and attaching the POP to resources defined in the Tivoli Access Manager object space (in the same way as the Tivoli Access Manager for Operating Systems product). The resource-level **permit** and **deny** audit levels can be tuned for more granular auditing of authorization access decisions, based on the action being performed against the protected resource.

Only resource-level auditing enabled for Sudo and Login resource types will result in the TAMOS Agent generating authorization audit records, since this is the only policy against which the TAMOS Agent makes an authorization decision.

Resource-level auditing enabled for File resources is supported by mapping this policy to native operating system audit settings. For more information about defining resource-level auditing for File resources, and how this is mapped to the native settings, see “Resource-level audit policy” on page 48. Once the policy is mapped, handled by the host operating systems’ own audit facilities. The TAMOS Agent does not generate audit records for accesses to File resources.

TAMOS Agent supports per-user level audit policy (**AuditAuth** and **AuditTrace**) by mapping it to native operating system settings. Per-user level audit policy does not enable the generation of TAMOS Agent audit records. For more information about the support of per-user audit policy, see “User-level audit policy” on page 52.

Warning mode is only supported for authorization decisions that the TAMOS Agent makes. The TAMOS Agent will only generate warning mode audit records for the Sudo and Login resource type, since this is the only policy against which the TAMOS Agent makes an authorization decision. Warning mode can be enabled by setting either the global warning mode or resource-level warning mode.

Operating System Files modified

The following operating system files are modified during the installation, configuration, and runtime of the TAMOS Agent. Files that are modified to reflect the installation itself are not listed. These lists assume a standard installation of the operating system.

Modifications made on AIX

The following operating system files are modified on AIX systems during the installation of TAMOS Agent:

- /etc/group
- /etc/passwd
- /etc/security/.ids
- /etc/security/group
- /etc/security/user

The following operating system files are modified during the configuration of TAMOS Agent:

- /etc/inittab
- /etc/security/user
- /etc/pam.conf

The following operating system files are modified when the TAMOS Agent Native Policy Manager maps policy to native settings:

- /etc/security/audit/objects - for resource-level audit policy,
- /etc/security/audit/config - for per-user audit policy and to add TAMOS Agent specific audit class definitions,
- /etc/security/user - for Login activity and password management policy.

Modifications made on Solaris

The following operating system files are modified on Solaris systems during the installation of TAMOS Agent:

- /etc/group
- /etc/passwd
- /etc/shadow

The following operating system files are modified during the configuration of TAMOS Agent:

- /etc/pam.conf

The following operating system files are modified when the TAMOS Agent Native Policy Manager maps policy to native settings:

- /etc/security/audit_control – for resource-level File audit policy,
- /etc/security/audit_user – for per-user audit policy
- /etc/security/policy.conf – for Login activity policy (lock after retries)
- /etc/default/passwd - for Login activity, Password management policy
- /etc/default/login - for Login activity policy (retry count)
- /etc/user_attr - for Login activity policy (lock after retries)
- /etc/shadow – for Login activity policy (min, max, inactive)

Note: On both AIX and Solaris, for the system files modified due to policy mapping, before the initial modification to each system file, the TAMOS Agent saves a copy of the original file. The copy is preserved under the name of the original file with .pdos.sav appended. For example, the backup of the /etc/security/audit_control is named /etc/security/audit_control.pdos.sav. You can use these backup files to restore the original system security settings after uninstalling the TAMOS Agent, or to examine the changes made by the TAMOS Agent after applying some policy.

You should not simply restore the backups to revert to the original settings, as this would lose any important changes that had been made (other than by the TAMOS Agent).

Tivoli Access Manager for Operating Systems 6.0 Management Tasks component not included

This component provides support for using the Tivoli Desktop to perform Tivoli Access Manager for Operating Systems management tasks, it is not provided with TAMOS Agent.

Chapter 5. Troubleshooting

This chapter describes known issues and limitations for the agent, and suggested steps for troubleshooting problems.

This chapter contains the following sections:

- “Known issues and limitations”
- “Pluggable Authentication Module Parameters” on page 70

Known issues and limitations

SWAP Filesystems do not support extended access control lists (ACLs)

On Solaris, the /tmp file system is often a swap file system. Swap file systems do not support extended ACLs. Therefore the TAMOS Agent will not be able to map access control policy to files or directories residing in a swap file system.

Globalization

The TAMOS Agent endpoint assumes that for policy branches to which the endpoint is configured, the defined policy is encoded in the same code page as the local code page on the system. For example, if policy is authored using UTF-8 multibyte character strings, the target TAMOS Agent systems using that policy must be running in the UTF-8 code page.

Solaris login activity policy for the root user

As a special case, the default login failure limit is not applied to the root user. The intention is to avoid locking the root user out of the system, which may be difficult to recover from. A login failure limit may be applied to the root user by creating a user exception in the policy specifically for root.

AIX GPFS™

The TAMOS Agent product does not support managing the access control policy for the AIX GPFS file system.

AIX Clogin and Solaris Zlogin Behavior

On AIX, a **clogin(1)** from the global WPAR to a non-global WPAR appears as a remote login to the non-global WPAR. Therefore, remote login location policy is applied.

On Solaris, a **zlogin(1)** from the global Zone to a non-global Zone appears as a local login to the non-global Zone. Therefore, local login location policy is applied.

System resource usage during policy application

When TAMOS Agent is processing a large number of policy objects (such as during policy reconciliation, after policy branch reconfiguration, or after policy changes that affect many files or users on the system), you may observe high CPU usage on 1 or 2 CPUs as well as I/O activity when updating file ACLs. This may continue for a few minutes, depending on the size of the policy and the machine specifications. On low end systems or systems with limited RAM or CPU resources, this may impact other applications running on the system

Pluggable Authentication Module Parameters

The TAMOS Agent Pluggable Authentication Module (PAM) has two parameters that can be useful for troubleshooting Login enforcement:

- **trace_string** and
- **allow_relative_hostnames**.

These parameters are set in the `/etc/pam.conf` file. The fifth and subsequent columns in the `pam.conf` file can be used to set parameters to pass to the module. The **pam_pdos_account** entry may appear multiple times in the `/etc/pam.conf` file, so it may be necessary to update these parameters in multiple places in order to have a consistent experience.

The **allow_relative_hostnames** parameter controls whether the login will be evaluated against the policy or immediately denied based on whether a relative hostname is provided to the TAMOS Agent PAM module. If this value is set to *yes*, remote login attempts that contain relative host names will not be automatically denied. Instead, the relative host name will attempt to be resolved and login policy will be evaluated. If the value is not set to *no*, or not set, login attempts that contain relative host names will be automatically denied. When TAMOS Agent module is added to the `/etc/pam.conf` file, this value is set to *no*.

The **trace_string** parameter is used to configure dynamic trace for the **pdosauthorize** utility, which is used to perform the authorization decision for the login event. The value of the trace string must be a valid trace routing entry. An example TAMOS Agent `/etc/pam.conf` entry that sets trace would be as follows:

```
other account requisite pam_pdos_account
allow_relative_hostnames=no
trace_string=out:pdosauthorize.9:FILE:/tmp/trace__other_pam.log
```

If a file is specified as the trace destination, the user that the login program is running as must have permission to write to the file specified.

Appendix A. Command reference

pdoscfg

Configures the TAMOS Agent.

Use the **pdoscfg** command to initially configure TAMOS Agent. After the initial configuration, use the **pdoscfg** command to modify the configuration attributes. The changes that are made with the **pdoscfg** command take effect the next time TAMOS Agent is started.

You can use the **pdoscfg** command to delete attributes from the configuration files. The daemons will use the default values the next time you start TAMOS Agent.

After the initial configuration, you cannot change the Active Directory domain with the **-AD_domain** option. This domain is required when the Tivoli Access Manager Runtime component is configured to multiple Active Directory domains. For TAMOS Agent, the Active Directory domain must be the same for all clients that are configured to the same Tivoli Access Manager policy server.

To re-configure the value of the **-admin_name**, **-admin_pwd**, **-branch**, **-local_domain**, or **-suffix** option, you must use the **pdosucfg** command to unconfigure TAMOS Agent. Then, use the **pdoscfg** command to re-configure the options with different values.

To re-configure the value of the **-ssl_listening_port**, **-registry_ssl_cacert**, or **-ldap_ssl_cacert** option, you must stop the TAMOS Agent before running the **pdoscfg** command.

Syntax

pdoscfg

```
[-AD_domain domain] [[-admin_cred_refresh minutes] [-admin_name admin_user_name] [-admin_pwd admin_user_password] [-audit_deny_actions osseal_action_group osseal_action_bits] [-audit_health health_types] [-audit_level audit_levels] [-audit_logflush seconds] [-audit_log_size bytes] [-audit_permit_actions osseal_action_group osseal_action_bits] [autostart {on|off}] -branch policy_branch [-certlife_interval days] [-certlife_threshold days] [-cred_hold minutes] [-cred_response_wait minutes] [-critical_cred_group group_name] [-critical_cred_refresh minutes] [-dns {on|off}] [-ffdc_capture {on|off}] [-heartbeat_interval minutes] [-hostname host_name] [-ldap_ssl_cacert ldap_certificate_name] [-local_domain management_domain] [-login_policy {on|off}] [-lrd_admin_name admin_user_name] [-lrd_admin_pwd admin_user_password] [-lrd_cars_ssl_add label] [-lrd_cars_ssl_cacert ssl_cacert_name] [-lrd_cars_ssl_delete label] [-lrd_cars_user_name_add cars_user_name] [-lrd_cars_user_name_delete cars_user_name] [-lrd_cars_user_name_pwd cars_user_password] [-lrd_config {on|off}] [-lrd_local_domain management_domain] [-pdosauditd_log_entries number_log_entries] [-pdosauditd_logs number_logs] [-pdosd_init_wait minutes] [-pdosd_log_entries number_log_entries] [-pdosd_logs number_logs] [-pdoslrd_log_entries number_log_entries] [-pdoslrd_logs number_logs] [-pdoswdd_log_entries number_log_entries] [-pdoswdd_logs number_logs] [-refresh_interval minutes] [-registry_ssl_cacert ssl_certificate_name] [-rspfile file_name] [-ssl_listening_port port] -suffix suffix [-uid {on|off}] [-user_cred_refresh minutes] [-verbose] [-warning {on|off}] ]
```

pdoscfg -delete options

```

pdoscfg -lrd_cars_ssl_list

pdoscfg -lrd_cars_user_name_list

pdoscfg -rspfile response_file

pdoscfg -help [ options]

pdoscfg -operations

pdoscfg -usage

pdoscfg -version

pdoscfg -V

pdoscfg -?

```

Parameters

Options for the configuration command are described in this section. The definition and default, if applicable, for each option is given.

-? Displays command usage information.

-AD_domain *domain*

Specifies the name of the Active Directory domain that contains the all of the users and groups used by TAMOS Agent. This option is required when configuring TAMOS Agent into a Tivoli Access Manager environment that uses multiple Active Directory domains. The specified domain must be the same for all TAMOS Agent clients that are configured to the same Tivoli Access Manager policy server. For all other situations, this option is not valid.

-admin_cred_refresh *minutes*

Specifies the refresh interval of administrator credentials in minutes. The default is 360.

-admin_name *admin_user_name*

Specifies the name of the Tivoli Access Manager administrator. The default is `sec_master`.

-admin_pwd *admin_user_password*

Specifies the password for the Tivoli Access Manager administrator. Use in combination with the **-admin_name** option. Replaces the **-sec_master_pwd** option.

-audit_deny_actions *osseal_action_group osseal_action_bits*

Specifies the [OSSEAL] action group followed by a list of the TAMOS Agent action bits to be audited. Valid action bits are `xL`. There is no default. To audit actions defined with this option, the **deny** and **logindeny** global audit levels must be defined in the configuration file or explicitly set with options of the **pdosctl** command for the current invocation of the daemon.

-audit_health *health_types*

Specifies a comma-separated list of audit health types. During a reconfiguration, the value specified for this option replaces the current value. The following values are valid:

- all
- certlife

- heartbeat
- none

The default is **certlife**.

-audit_level *audit_levels*

Specifies a comma-separated list of audit levels. The following values are valid:

- admin
- all
- deny
- logindeny
- loginpermit
- info
- none
- permit
- trace_file
- verbose

The default is **none**. Audit levels specified here only take effect for **login** and **pdossudo** operations.

-audit_logflush *seconds*

Specifies the interval in seconds that the **pdosauditd** daemon flushes the audit records to the active audit log. The default is 5.

-audit_log_size *bytes*

Specifies the maximum size in bytes to which the active audit log can grow before the **pdosauditd** daemon rolls over to use a new active audit log. The default is 1000000.

-audit_permit_actions *osseal_action_group osseal_action_bits*

Specifies the [OSSEAL] action group followed by a list of the TAMOS Agent action bits to be audited. Valid action bits are **xL**. There is no default. To audit actions defined with this option, the **permit** and **loginpermit** global audit levels must be defined in the configuration file or explicitly set with options of the **pdosctl** command for the current invocation of the daemon.

-autostart {on | off}

Indicates whether to automatically start TAMOS Agent at system reboot. The default is *on*.

-branch *policy_branch*

Specifies the name of the initial policy branch to which this machine subscribes. There is no default.

-certlife_interval *days*

Specifies the interval in days between the generation of certificate lifetime audit records.

Note: To generate certificate lifetime audit records, certificate lifetime health auditing must be enabled (the value for the **-audit_health** option must include **certlife**). When not enabled, the value specified for the **-certlife_interval** option is written to the configuration file. After enabling certificate lifetime health auditing, this saved value will be used.

- If the **-certlife_threshold** option is set to a nonzero value, the interval for the generation of certificate lifetime audit records becomes daily when the certificate threshold is reached.
- If the **-certlife_interval** option is set to zero, no certificate lifetime audit records is generated until the certificate threshold is reached. After reaching this threshold, the interval for the generation of certificate lifetime audit records becomes daily.
- If both the **-certlife_interval** option and the **-certlife_threshold** option are set to zero, no certificate lifetime audit records are generated.
- The default is 7.

-certlife_threshold *days*

Specifies the threshold in days, after which the generation of certificate lifetime audit records and the generation of certificate lifetime warning messages becomes daily.

Note: To generate certificate lifetime audit records, certificate lifetime health auditing must be enabled (the value for the **-audit_health** option must include **certlife**). When not enabled, the value specified for the **-certlife_threshold** option is written to the configuration file. After enabling certificate lifetime health auditing, this saved value will be used.

- If the **-certlife_threshold** option is less than or equal to the time before the certificate expires, certificate lifetime audit records are generated and certificate expiration warning messages are written daily to the daemon log file. After refreshing the certificate, the value specified for the **-certlife_interval** option controls the interval at which certificate lifetime audit records are generated.
- If the **-certlife_threshold** option is set to zero, certificate lifetime audit records are generated according to the value specified for the **-certlife_interval** option. Certificate expiration warning message are not written to the daemon log file.
- If both the **-certlife_threshold** option and the **-certlife_interval** option are set to zero, no certificate lifetime audit records are generated.
- The default is 30.

-cred_hold *minutes*

Specifies the maximum amount of time in minutes that a non-administrator credential is cached without being accessed. This value must be greater than or equal to the values of the **-admin_cred_refresh** option and the **-user_cred_refresh** option. The default is 10080.

-cred_response_wait *minutes*

Specifies the minimum length of time in minutes to wait for a response to a credential request before entering isolation mode. The default is 2.

-critical_cred_group *group_name*

Specifies the name of the Tivoli Access Manager group whose members are to be treated as critical system users and whose credentials should always be available in the credential cache. There is no default.

-critical_cred_refresh *minutes*

Specifies the refresh interval for the credentials of the critical system users in minutes. These users are members of the group defined by the **-critical_cred_group** option). The default is 720.

-delete options

Specifies a comma-separated list of options to remove from configuration files. The following values are valid:

- admin_cred_refresh
- audit_level
- audit_log_entries
- audit_logflush
- audit_logs
- audit_log_size
- audit_deny_actions
- audit_permit_actions
- certlife_interval
- certlife_threshold
- cred_hold
- cred_response_wait
- critical_cred_group
- critical_cred_refresh
- dns
- ffdc_capture
- heartbeat_interval
- pdosd_log_entries
- pdosd_logs
- pdoswdd_log_entries
- pdoswdd_logs
- refresh_interval
- uid
- user_cred_refresh
- warning

-dns {on | off}

Indicates whether to enable TAMOS Agent to store the IP address to host name mapping information. The default is *on*.

-ffdc_capture {on | off}

Indicates whether to enable the capture of first failure data upon abnormal termination of the core TAMOS Agent daemons. The default is *on*.

-heartbeat_interval minutes

Specifies the interval in minutes between the generation of heartbeat audit events.

Note: To generate heartbeat audit events, heartbeat auditing must be enabled (the value for the **-audit_health** option must include **heartbeat**). When not enabled, the value specified for the **-heartbeat_interval** option is written to the configuration file. After enabling heartbeat auditing, this saved value will be used. The default is *10*.

-help option

Displays help for all of the options. To display help for one option, use **-help** option. For example, to display help for the **-audit_level** option, enter **-help -audit_level**.

-hostname *host_name*

Specifies the host name that will be used by the Tivoli Access Manager server to recognize this machine. If not specified, the default is the local host name returned by the operating system.

-ldap_ssl_cacert *ldap_certificate_name*

This option is deprecated and replaced with the **-registry_ssl_cacert** option.

-local_domain *management_domain*

Specifies the Tivoli Access Manager secure domain into which the **pdosd** daemon is to be configured. If this option is not specified, the local domain defaults to the secure domain that the Tivoli Access Manager Runtime configuration is using. If a domain was not specified when the Tivoli Access Manager Runtime component was configured, its local domain defaulted to the Default management domain. The Tivoli Access Manager secure domain must exist and the administrator name and password specified with the **-admin_name** and **-admin_pwd** options must be valid for this domain.

-login_policy {on | off}

Indicates whether to enable system login and password restrictions. After enabling login policy, any of the graphical login methods, such as **dtlogin**, that are running must be restarted if login activity policy is to be active for logins that use those methods. When the graphical login program is restarted, the login activity policy is read and made active. The default is *on*.

-lrd_admin_name *admin_user_name*

Specifies the Tivoli Access Manager administrator name to use when registering the **pdoslrd** daemon with the Tivoli Access Manager policy server.

-lrd_admin_pwd *admin_user_password*

Specifies the Tivoli Access Manager administrator password to use when registering the **pdoslrd** daemon with the Tivoli Access Manager policy server.

-lrd_cars_ssl_add *label*

Specifies the label of the CA certificate to add for the specified Common Auditing and Reporting Service event server.

-lrd_cars_ssl_cacert *ssl_cacert_name*

Specifies the file name of the CA certificate of the Common Auditing and Reporting Service event server that receives TAMOS Agent auditing events. This certificate is required for the mutual authentication that occurs between TAMOS Agent and Common Auditing and Reporting Service event server and for the encryption of audit records that are sent to the Common Auditing and Reporting Service event server.

-lrd_cars_ssl_delete *label*

Specifies the label of the CA certificate to delete for the specified Common Auditing and Reporting Service event server.

-lrd_cars_ssl_list

Displays the list of CA certificates for all defined Common Auditing and Reporting Service event servers.

-lrd_cars_user_name_add *cars_user_name*

Specifies the user name to add for authenticating with Common Auditing and Reporting Service event servers. This option, along with the **-lrd_cars_user_name_pwd**, is required if the event server is configured to authenticate with the client.

- lrd_cars_user_name_delete** *cars_user_name*
Specifies the user name to delete. After being deleted, that user cannot be used to authenticate with Common Auditing and Reporting Service event servers.
- lrd_cars_user_name_list** *cars_user_name*
Lists the user names that are associated with Common Auditing and Reporting Service event servers and were configured using the **-lrd_cars_user_name_add** option.
- lrd_cars_user_name_pwd** *cars_user_password*
Specifies the password for the user that is used to authenticate with the Common Auditing and Reporting Service event servers.
- lrd_config {on | off}**
Indicates whether to configure or unconfigure the **pdoslrd** daemon. The default is *off*.
- lrd_local_domain** *management_domain*
Specifies the name of the Tivoli Access Manager secure domain that the **pdoslrd** daemon will be configured to use. If the **pdoslrd** daemon will be used to send audit data to a Tivoli Access Manager authorization server (**pdacld** daemon) as a remote collection point, the **pdoslrd** daemon must be configured into the same secure domain that the **pdacld** daemon is configured to use. In an environment where the Tivoli Access Manager policy server is managing multiple secure domains, this might mean that the **pdoslrd** daemon needs to be configured into a different secure domain than the **pdosd** daemon. If this option is not specified, the local domain defaults to the secure domain that the **pdosd** configuration is using. This Tivoli Access Manager secure domain must exist and the administrator name and password specified with the **-lrd_admin_name** and **-lrd_admin_pwd** options must be valid for this domain.
- operations**
Lists the supported options in a condensed format.
- pdosauditd_log_entries** *number_log_entries*
Specifies the number of log entries to write before archiving the **pdosauditd** log file. If the values for the **-pdosauditd_log_entries** option and the **-pdosauditd_logs** option are nonzero, the **pdosauditd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdosauditd_log_entries** option or when the **pdosauditd** daemon is restarted. If the value for the **-pdosauditd_log_entries** option is nonzero and the value for the **-pdosauditd_logs** option is zero, the **pdosauditd** log file will be recycled when the number of entries reaches the number specified by the **-pdosauditd_log_entries** option or when the **pdosauditd** daemon is restarted. The default of 0 means that the number of entries to write is unlimited and that the log file will not be archived.
- pdosauditd_logs** *number_logs*
Specifies the number of archive log files to use before recycling the **pdosauditd** archive log files. Setting the number of archive log files to a nonzero value has an effect only if the value for the **-pdosauditd_log_entries** option is nonzero. The **pdosauditd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdosauditd_log_entries** option or when the **pdosauditd** daemon is restarted. The default of 0 means never archive the **pdosauditd** log file.

-pdosd_init_wait *minutes*

Specifies the maximum number of minutes to wait at startup for the background **pdosd** daemon to complete initialization and enable policy enforcement. The default is 5.

-pdosd_log_entries *number_log_entries*

Specifies the number of log entries to write before archiving the **pdosd** log file. If the values for the **-pdosd_log_entries** option and the **-pdosd_logs** option are nonzero, the **pdosd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdosd_log_entries** option or when the **pdosd** daemon is restarted. If the value for the **-pdosd_log_entries** option is nonzero and the value for the **-pdosd_logs** option is zero, the **pdosd** log file will be recycled when the number of entries reaches the number specified by the **-pdosd_log_entries** option or when the **pdosd** daemon is restarted. The default of 0 means that the number of entries to write is unlimited and that the log file will not be archived.

-pdosd_logs *number_logs*

Specifies the number of archive log files to use before recycling the **pdosd** archive log files. Setting the number of archive log files to a nonzero value has an effect only if the value for the **-pdosd_log_entries** option is nonzero. The **pdosd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdosd_log_entries** option or when the **pdosd** daemon is restarted. The default of 0 means never archive the **pdosd** log file.

-pdoslpmd_failure_audit_seconds *seconds* Specifies the maximum amount of time in seconds for auditing login failures. The default value is 10.

-pdoslrd_log_entries *number_log_entries*

Specifies the number of log entries to write before archiving the **pdoslrd** log file. If the values for the **-pdoslrd_log_entries** option and the **-pdoslrd_logs** option are nonzero, the **pdoslrd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdoslrd_log_entries** option or when the **pdoslrd** daemon is restarted. If the value for the **-pdoslrd_log_entries** option is nonzero and the value for the **-pdoslrd_logs** option is zero, the **pdoslrd** log file will be recycled when the number of entries reaches the number specified by the **-pdoslrd_log_entries** option or when the **pdoslrd** daemon is restarted. The default of 0 means that the number of entries to write is unlimited and that the log file will not be archived.

-pdoslrd_logs *number_logs*

Specifies the number of archive log files to use before recycling the **pdoslrd** archive log files. Setting the number of archive log files to a nonzero value has an effect only if the value for the **-pdoslrd_log_entries** option is nonzero. The **pdoslrd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdoslrd_log_entries** option or when the **pdoslrd** daemon is restarted. The default of 0 means never archive the **pdoslrd** log file.

-pdoswdd_log_entries *number_log_entries*

Specifies the number of log entries to write before archiving the **pdoswdd** log file. If the values for the **-pdoswdd_log_entries** option and the **-pdoswdd_logs** option are nonzero, the **pdoswdd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdoswdd_log_entries** option or when the **pdoswdd** daemon is restarted. If the value for the **-pdoswdd_log_entries** option is nonzero and the value for the **-pdoswdd_logs** option is zero, the **pdoswdd** log file will be recycled when the number of entries reaches the number specified by the

-pdoswdd_log_entries option or when the **pdoswdd** daemon is restarted. The default of 0 means that the number of entries to write is unlimited and that the log file will not be archived.

-pdoswdd_logs *number_logs*

Specifies the number of archive log files to use before recycling the **pdoswdd** archive log files. Setting the number of archive log files to a nonzero value has an effect only if the value for the **-pdoswdd_log_entries** option is nonzero. The **pdoswdd** log file will be archived when the number of entries reaches the number of entries specified by the **-pdoswdd_log_entries** option or when the **pdoswdd** daemon is restarted. The default of 0 means never archive the **pdoswdd** log file.

-refresh_interval *minutes*

Specifies the interval in minutes that the Tivoli Access Manager policy server is polled for policy database updates, if it has not received updates during this interval. A value of zero indicates that policy database updates are not received by polling. Compare with the **-ssl_listening_port** option. The default is 0.

-registry_ssl_cacert *ssl_certificate_name*

Specifies the SSL certificate file of the Tivoli Access Manager user registry. This certificate is required for the mutual authentication that occurs between TAMOS Agent and the Tivoli Access Manager user registry server. If the **install_ldap_server** program was used to install and configure IBM Tivoli Directory Server and you chose to use the default SSL CA certificate file that was provided by Tivoli Access Manager, you must obtain the `/etc/gsk/pd_ldapcert.arm` file from the user registry server and use that file during the configuration of TAMOS Agent.

-rspfile *file_name*

Specifies the name of a file that contains option values for the configuration.

-ssl_listening_port *port*

Specifies the port to listen for policy database update notifications. A value of zero indicates that policy database updates will not be received by notification. Compare with the **-refresh_interval** option. The default is 7134.

-suffix *suffix*

Specifies the user registry suffix under which the Tivoli Access Manager users and groups for TAMOS Agent should be created during configuration. An example suffix is `ou=austin,o=ibm,c=us`. If the suffix contains spaces, enclose it in quotation marks ("*suffix*").

-uid {on | off}

Indicates whether to enable caching of the UID/GID to user/group name-mapping information. The default is *off*.

-usage

Displays command usage information.

-user_cred_refresh *minutes*

Specifies the refresh interval of user credentials in minutes. The default is 720.

-verbose

Displays verbose messages.

-version

Displays the version of the command.

-warning {on | off}

Indicates whether to enable global authorization warning mode. The default is *off*.

Authorization

You must be a TAMOS Agent administrator that has privileges to create or modify groups and users to use this command.

Examples

1. To configure an TAMOS Agent client to the AServers policy branch from the command line, enter:

```
pdoscfg -admin_name sec_master -admin_pwd password \ -registry_ssl_cacert  
/tmp/ldapcacert.b64 -branch AServers \ -suffix o=tivoli
```

2. To configure an TAMOS Agent client from the /opt/pdos/etc/config.rsp response file and override the values for the **-uid** and **-audit_level** options, enter:

```
pdoscfg -rspfile /opt/pdos/etc/config.rsp \ -uid off -audit_level all
```

pdosucfg

Unconfigures the TAMOS Agent.

Use the **pdosucfg** command to unconfigure TAMOS Agent. After unconfiguration, TAMOS Agent may be reconfigured using the **pdoscfg** command. TAMOS Agent must be unconfigured before it can be uninstalled from the system.

Syntax

pdosucfg

[--admin_name *admin_user_name*] **[--admin_pwd** *admin_user_password*] **[--force]**
[--lrd_admin_name *admin_user_name*] **[--lrd_admin_pwd** *admin_user_password*]
[--remove_once_only] **[--remove_per_policy]** **[--rspfile** *file_name*]

pdosucfg --help

pdosucfg --usage

pdosucfg --version

pdosucfg -V

pdosucfg -?

Parameters

Options for the unconfiguration command are described in this section. The definition and default, if applicable, for each option is given.

-? Displays command usage information.

--admin_name *admin_user_name*

Specifies the name of the Tivoli Access Manager administrator. The default is *sec_master*.

--admin_pwd *admin_user_password*

Specifies the password for the Tivoli Access Manager administrator. Use in combination with the **--admin_name** option. Replaces the **--sec_master_pwd** option.

--lrd_admin_name *admin_user_name*

Specifies the Tivoli Access Manager administrator name to use when registering the **pdoslrd** daemon with the Tivoli Access Manager policy server.

--lrd_admin_pwd *admin_user_password*

Specifies the Tivoli Access Manager administrator password to use when registering the **pdoslrd** daemon with the Tivoli Access Manager policy server.

--remove_once_only

If specified, **pdosucfg** will unregister the IBM Tivoli Access Manager for Operating Systems product policy.

--remove_per_policy

If specified, **pdosucfg** will unregister the policy-specific policy information.

--rspfile *file_name*

Specifies the name of a file that contains option values for the configuration.

-version

Displays the version of the command.

Authorization

You must be a TAMOS Agent administrator that has privileges to create or modify groups and users to use this command.

Appendix B. Support information

This section describes the following options for obtaining support for IBM products:

- “Searching knowledge bases”
- “Obtaining fixes”
- “Registering with IBM Software Support” on page 86
- “Receiving weekly software updates” on page 86
- “Contacting IBM Software Support” on page 87

Searching knowledge bases

If you encounter a problem, you want it resolved quickly. You can search the available knowledge bases to determine whether the resolution to your problem was already encountered and is already documented.

Searching information centers

IBM provides extensive documentation in an information center that can be installed on your local computer or on an intranet server. You can use the search function of this information center to query conceptual information, instructions for completing tasks, reference information, and support documents.

Searching the Internet

If you cannot find an answer to your question in the information center, search the Internet for the latest, most complete information that might help you resolve your problem. To search multiple Internet resources for your product, perform the following steps:

1. Expand the product folder in the navigation frame on the left.
2. Expand **Troubleshooting and support**.
3. Expand **Searching knowledge bases**.
4. Click **Web search**.

From this topic, you can search a variety of resources, which includes the following resources:

- IBM Technotes
- IBM downloads
- IBM Redbooks
- IBM developerWorks®
- Forums and news groups
- Google

Obtaining fixes

A product fix might be available to resolve your problem. To determine what fixes are available for your IBM software product, check the product support site by performing the following steps:

1. Go to the IBM Software Support site at the following Web address:

<http://www.ibm.com/software/support>

2. Under **Products A - Z**, click the letter with which your product starts to open a Software Product List.
3. Click your product name to open the product-specific support page.
4. Under **Self help**, follow the link to **All Updates**, where you will find a list of fixes, fix packs, and other service updates for your product. For tips on refining your search, click **Search tips**.
5. Click the name of a fix to read the description.
6. Optional, download the fix.

Registering with IBM Software Support

Before you can receive weekly e-mail updates about fixes and other news about IBM products, you need to register with IBM Software Support. To register with IBM Software Support, follow these steps:

1. Go to the IBM Software Support site at the following Web address:

<http://www.ibm.com/software/support>

2. Click **Register** in the upper right-hand corner of the support page to establish your user ID and password.
3. Complete the form, and click **Submit**.

Receiving weekly software updates

After registering with IBM Software Support, you can receive weekly e-mail updates about fixes and other news about IBM products. To receive weekly notifications, follow these steps:

1. Go to the IBM Software Support site at the following Web address

<http://www.ibm.com/software/support>

2. Click the **My support** link to open the Sign in page.
3. Provide your sign in information, and click **Submit** to open your support page.
4. Click the **Edit profile** tab.
5. For each product about which you want to receive updates, use the filters to choose your exact interests, and click **Add products**.
6. Repeat step 5 for each additional product.
7. After choosing all your products, click the **Subscribe to email** link.
8. For each product category, use the filters and choose which updates you want to receive, and click **Update**.
9. Repeat step 8 for each additional product category.

For more information about the types of fixes that are available, see the *IBM Software Support Handbook* at the following Web address:

<http://techsupport.services.ibm.com/guides/handbook.html>

Contacting IBM Software Support

IBM Software Support provides assistance with product defects. Before contacting IBM Software Support, the following criteria must be met:

- Your company has an active IBM software maintenance contract.
- You are authorized to submit problems to IBM Software Support.

The type of software maintenance contract that you need depends on the type of product that you have. Product types are one of the following categories:

- For IBM distributed software products (including, but not limited to, Tivoli, Lotus®, and Rational® products, as well as DB2 and WebSphere products that run on Windows, Linux®, or UNIX operating systems), enroll in Passport Advantage® in one of the following ways:

Online

Go to the IBM Software Passport Advantage site at the following Web address and click **How to Enroll**:

[http://www.lotus.com/services/passport.nsf/
WebDocs/Passport_Advantage_Home](http://www.lotus.com/services/passport.nsf/WebDocs/Passport_Advantage_Home)

By phone

For the phone number to call in your country, go to the IBM Software Support site at the following Web address and click the name of your geographic region:

<http://techsupport.services.ibm.com/guides/contacts.html>

- For IBM eServer™ software products (including, but not limited to, DB2 and WebSphere products that run in System z®, pSeries®, and iSeries® environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM eServer Technical Support Advantage site at the following Web address:

<http://www.ibm.com/servers/eserver/techsupport.html>

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States or, from other countries, go to the contacts page of the *IBM Software Support Handbook* at the following Web address and click the name of your geographic region for phone numbers of people who provide support for your location:

<http://techsupport.services.ibm.com/guides/contacts.html>

To contact IBM Software support, follow these steps:

1. "Determining the business impact"
2. "Describing problems and gathering information" on page 88
3. "Submitting problems" on page 88

Determining the business impact

When you report a problem to IBM, you are asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem that you are reporting. Use the following severity criteria:

Severity 1

The problem has a *critical* business impact. You are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.

Severity 2

The problem has a *significant* business impact. The program is usable, but it is severely limited.

Severity 3

The problem has *some* business impact. The program is usable, but less significant features that are not critical are unavailable.

Severity 4

The problem has *minimal* business impact. The problem causes little impact on operations, or a reasonable circumvention to the problem was implemented.

Describing problems and gathering information

When explaining a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Software Support specialists can help you solve the problem efficiently. To save time, know the answers to these questions:

- What software versions were you running when the problem occurred?
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
- Can you create the problem again? If so, what steps were performed to encounter the problem?
- Was any change made to the system? For example, were there changes to the hardware, operating system, networking software, and so on.
- Are you currently using a workaround for this problem? If so, please be prepared to explain it when you report the problem.

Submitting problems

You can submit your problem to IBM Software Support in one of two ways:

Online

Go to the Submit and track problems page on the IBM Software Support site at the following address, and provide your information into the appropriate problem submission tool:

<http://www.ibm.com/software/support/probsub.html>

By phone

For the phone number to call in your country, go to the contacts page of the *IBM Software Support Handbook* at the following Web address and click the name of your geographic region:

<http://techsupport.services.ibm.com/guides/contacts.html>

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Software Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the IBM product support Web pages daily, so that other users who experience the same problem can benefit from the same resolution.

For more information about problem resolution, see “Searching knowledge bases” on page 85 and “Obtaining fixes” on page 85.

Appendix C. Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Trademarks

IBM, the IBM logo, AIX, DB2, Tivoli, and Tivoli Enterprise Console are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Adobe, Acrobat, PostScript® and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.



Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT®, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.



Printed in USA

SC23-9802-01

