IBM Security Verify Access
Version 10.0.8
June 2024

Command Reference topics

Contents

Tables

Chapter 1. pdadmin commands

The **pdadmin** command-line utility is installed as part of the IBM Security Verify Access runtime package.

Use this interface to manage access control lists, groups, servers, users, objects, and other resources in your secure domain. You can also automate certain management functions by writing scripts that use **pdadmin** commands.

Use the Web Portal Manager interface to complete remotely similar administrative tasks. When you use Web Portal Manager, no special network configuration is needed to connect and complete these management tasks.

How to read syntax statements

Syntax diagrams pictorially display the order and parameters for the command utility.

The reference documentation uses the following special characters to define syntax:

[ ]
: Identifies optional options. Options that are not enclosed in brackets are required.

...
: Indicates that you can specify multiple values for the previous option.

|
: Indicates mutually exclusive information. You can use the option to the left of the separator or the option to the right of the separator. You cannot use both options in a single use of the command.

{ }
: Delimits a set of mutually exclusive options when one of the options is required. If the options are optional, they are enclosed in brackets ([ ]).

****
: Indicates that the command line wraps to the next line. It is a continuation character.

The options for each command are listed alphabetically in the Options section. The options for each utility are listed alphabetically in the Parameters section. When the order of the options or parameters must be used in a specific order, this order is shown in the syntax statements.

Syntax for pdadmin commands

The following syntax is used with the **pdadmin** command:

pdadmin [–I *configuration-instance-name*] [[–a *admin_id* [–p *password*] [–d *domain*]] [–linelen *max-linelen*] [–histsize *history size*] [–v] [*command*]

pdadmin [–I *configuration-instance-name*] [[–a *admin_id* [–p *password*] [–d *domain*]] [–linelen *max-linelen*] [–v] [*file*]

pdadmin [–I *configuration-instance-name*] [[–a *admin_id* [–p *password*] [–m]] [–linelen *max-linelen*] [–v] [*command*]

pdadmin [–I *configuration-instance-name*] [[–a *admin_id* [–p *password*] [–m]] [–linelen *max-linelen*] [–v] [*file*]

pdadmin [–l] [–linelen *max-linelen*] [–*v*] [*command*]

pdadmin [–l] [–linelen *max-linelen*] [–v] [*file*]

The following list explains the options for the **pdadmin** utility:

command
: Specifies the single **pdadmin** command to run. The command is run one time. The **pdadmin** utility does not enter interactive mode. The *command* option is mutually exclusive with the *file* option.

file

Specifies the fully qualified name of the file that contains a list of commands to run. These commands are run one time. The **pdadmin** utility does not enter interactive mode. The *file* option is mutually exclusive with the *command* option.

Note: For Windows operating systems, file names cannot contain the backward slash (\), colon (:), question mark (?), or double quotation mark characters.

–a *admin_id*

Logs you in as the user *admin_id*. This administrator must exist in the domain. If you do not specify this option on the command line, you are considered unauthenticated, and your access to other commands is limited. If you specify this option without specify the –p option, you are prompted for the password.

The –a option is mutually exclusive with the –l option. If you do not specify either option, you are logged in as an unauthenticated user. Unauthenticated users can use the **context**, **errtext**, **exit**, **help**, **login**, **logout** and **quit** commands only.

–d *domain*

Specifies the Security Verify Access secure domain to log in. Log in to this domain requires authentication. The *admin_id* user that is specified must exist in this domain. The –d option is mutually exclusive with the –m option. If neither options are specified, the target domain is the local domain that is configured for the system.

–I *configuration-instance-name*

Specifies the pd.conf file instance that the **pdadmin** command should use. The *configuration-instance-name* value is the *hostname* that is provided to the **pdadmin_host** command that generated the configuration file. This option allows **pdadmin** to communicate with multiple policy servers.

–l

Specifies a local login operation. When modifications are made to local configuration files by using the **pdadmin config** commands, a local login is required before you can run commands.

The –l option is mutually exclusive with the –a option. If you do not specify either option, you are logged in as an unauthenticated user. Unauthenticated users can use the **context**, **errtext**, **exit**, **help**, **login**, **logout** and **quit** commands only.

–linelen *max-linelen*

Currently, the –linelen option is ignored.

–m

Specifies that the login operation must be directed to the management domain. Log in to this domain requires authentication. The *admin_id* user that is specified must exist in this domain. The –m option is mutually exclusive with the –d option. If neither options are specified, the target domain is the local domain that is configured for the system.

–p *password*

Specifies the password for the user *admin_id*. Using this option might show your password to others because the password is visible on the screen and also in the process table. If you do not specify this option on the command line, you are prompted for a password. This option cannot be used unless the –a option is used.

–v

Prints the version number of the **pdadmin** utility. If this option is specified, all other valid options are ignored.

The following example is the output that you might see when you use this option:

```
Security Verify Access Administrative Tool v10.0.0.0 (Build 20200202)
Copyright (C) IBM Corporation 1994-2020. All Rights Reserved.
```

-histsize
Specifies the command history size. The default command history size is 64. The minimum size of the command history is 1 and the maximum size is 1024. The command history option is available only in the interactive mode and on operating systems other than Windows.

Note:

1. If you specify the –a and –p options, you are logged in as that user. Using this method might show your password to others. For example, one user is using **pdadmin** with this command. Another user lists the processes that are running. Then, the full command that includes the password, might be visible to the second user.
2. Users can run the **pdadmin context show** command to view their authentication information.

Command modes

You can use the **pdadmin** utility in three different command modes: single, interactive, or multiple.

These modes are described in the following sections.

For details about the command options that are displayed in the following sections, see "Syntax for pdadmin commands" on page 1.

Single command mode

In single command mode, the CLI runs only the specified command, and ends after it receives the response message for that command.

To run a single **pdadmin** command, enter one of the following commands:

pdadmin [–a *admin_id* [–p *password*] [–d *domain*]] [–v] [*command*]

pdadmin [–a *admin_id* [–p *password*] [–m]] [–v] [*command*]

pdadmin [–l] [–v] [*command*]

For details about the command options, see "Syntax for pdadmin commands" on page 1.

Interactive command mode

Interactive command mode uses an interactive command-line session where, after the command starts, you are prompted to enter required information.

To start **pdadmin** in interactive mode, type the **pdadmin** command.

This command starts **pdadmin** without any authentication that is required, where your access to other **pdadmin** commands is limited for unauthenticated users, such as **context**, **errtext**, **exit**, **help**, **login**, **logout**, and **quit**.

```
c:\> pdadmin

pdadmin> limited_pdadmin_command
```

This command starts **pdadmin** and login authentication is required before you can use other **pdadmin** commands. You can be prompted for both the administrator ID and the password:

```
c:\> pdadmin

pdadmin> login
Enter User ID:sec_master
Enter Password: secmstrpw

pdadmin sec_master> pdadmin_command
```

Or, you can be prompted for just the administrator password:

```
c:\> pdadmin
```

```
pdadmin> login -a sec_master
Enter Password: secmstrpw

pdadmin sec_master> pdadmin_command
```

Or, you can bypass being prompted, which is less secure because your password is visible:

```
c:\> pdadmin

pdadmin> login -a sec_master -p secmstrpw

pdadmin sec_master> pdadmin_command
```

To start **pdadmin** in interactive mode:

- With a login to a management or other domain.
- Where the ID and password are authenticated before access is permitted.
- Where user privileges are verified before users can issue commands.

For example, to log in to the management domain (`Default`) and authenticate, type:

```
pdadmin login -a admin_id -p password -m

pdadmin sec_master@Default> pdadmin_command
```

For example, to log in to another domain `domain01` and authenticate, type:

```
rpdadmin login -a sec_master -p secmstrpw -d domain01

pdadmin sec_master@domain01> pdadmin_command
```

At the **pdadmin** prompt, type the appropriate commands and their associated options. The **pdadmin** prompt changes, depending on the type of login. See "Login and logout commands" on page 10 for more information about the **login** command and prompt changes.

Note: In this release, the length of a command line that is used in **pdadmin** interactive mode is limited to 1023 characters.

Multiple command mode

With multiple command mode, you can create a file that contains multiple **pdadmin** commands, one per line, that together complete a task or series of tasks.

Login commands can be included in a command file to switch between local and remote login, as needed.

To run commands in this file, provide one of the following commands:

pdadmin [–a *admin_id* [–p *password*] [–d *domain*]] *file*

pdadmin [–a *admin_id* [–p *password*] [–m]] *file*

Login commands can be included in a command file to switch between **pdadmin login -l** local login:

- Where no authentication is required.
- Where authentication is required.

For details about the command options that are displayed in the following sections, see "Syntax for pdadmin commands" on page 1.

Non-English locales

For Security Verify Access software, you can specify localized behavior by setting the required locale.

Different operating systems often encode text in different ways. For example, Windows operating systems use SJIS (code page 932) for Japanese text while AIX, Linux, and Solaris operating systems often use eucJP.

However, be aware of the following issues when you are running the **pdadmin** utility in a non-English locale.

- On Windows operating systems, you can enter commands to **pdadmin** through a command file argument. The command file must be encoded in the system's local (ANSI) code page. For example:

  ```
  C:> pdadmin -a sec_master -p password cmds.text
  ```

  You can determine the local code page of the system by viewing the value of the **Nls/CodePage/ACP** key in the Windows registry. Files that are created by standard Windows editing tools (such as Notepad or WordPad) are encoded in this way.

  On AIX, Linux, and Solaris operating systems, you must run the **pdadmin** command in the same locale that was used to create the command file.

- On Windows operating systems, you can enter commands to **pdadmin** by redirecting a command file. The command file must be encoded in a Microsoft Original Equipment Manufacturer (OEM) code page. The OEM code page corresponds to the active code page in the command window in which the **pdadmin** command is run. For example:

  ```
  C:> pdadmin -a sec_master -p password < cmds.text
  ```

  The active code page can be determined by issuing the **chcp** command in the **pdadmin** command window.

  Alternatively, you can redirect a file that is encoded in the local code page of the system. However, you must change the active code page of the command window to correspond to the encoding of the file. Change the active code page of the window by using the **chcp** command. For example, entering the command `chcp 1252` changes the active code page to the ANSI code page for Western Europe and the United States.

  On AIX, Linux, and Solaris operating systems, you must run the **pdadmin** command in the same locale that was used to create the redirected command file.

- Security Verify Access data that is created in one locale might not display correctly on a system that is configured to another locale. Whether data displays correctly depends on the configuration of the second system. For example, correct display depends on what the current locale is, and whether the necessary code pages and fonts are installed.

Error handling

After a command finishes processing, a return code is displayed or logged to provide the success or failure of the command.

The **pdadmin** command has the following return code values:

0

The command that completed successfully.

1

The command failed. When a command fails, the **pdadmin** command displays a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM® Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

For information about how to use the message number that is associated with a message to display only the descriptive text, see "errtext" on page 44.

Return codes for a single command

A single command is normally typed from a command prompt such as a DOS command prompt, Korn shell prompt, and C shell prompt. Single command mode does not automatically display the `0` or `1` return code values; the operating system must be queried for the return code value.

For command failures, the hexadecimal error code status with its associated error message is shown in addition to the error message ID (for example, `HPDMG0754W`). You can redirect the error that is normally displayed on the screen out to a text file. When a single command fails, you see an error message that is like the one displayed:

```
C:> pdadmin -a admin_id -p password user show oogle
```

```
Could not perform the administration request.
Error: HPDMG0754WThe entry was not found. If ...
(status 0x14c012f2)
```

To display the `0` or `1` return code values, you must type the **pdadmin** command, followed by either the AIX®, Linux®, or Solaris **echo** or the Windows **`errorlevel`** command:

- For AIX, Linux, and Solaris operating systems:

  ```
  # pdadmin_command
  # echo $?
  ```

- For Windows operating systems:

  ```
  C:>pdadmin_command
  C:>echo %errorlevel%
  ```

Return codes for an interactive command

Interactive command mode does not automatically display the `0` or `1` return code values. Also, you cannot follow an interactive command with the AIX, Linux, and Solaris **echo** or the Windows **`errorlevel`** command.

For a command failure, you see a message that is like:

```
pdadmin sec_master> user show oogle
```

```
Could not perform the administration request.
Error: HPDMG0754WThe entry was not found. If ...
(status 0x14c012f2)
```

Only the hexadecimal exit status code is displayed.

Return codes for multiple commands

You can use a text file containing **`pdadmin`** commands to run those commands in a single **`pdadmin`** invocation.

Consider that an error occurs for a command while the commands run in multiple command mode. Then, an error message for the failed command is provided.

Processing of the remaining commands in the file continues after an error. At the end of multiple command processing, a final status is provided. The final status code at the termination of multiple command processing is only for the last command that was attempted. For example, if the last command was successful, the final status is `0`. If the last command failed, the final status is `1`.

For example, a text file might contain these **`pdadmin`** commands:

```
user show cwright
user show oogle
```

To run the commands, run the following command:

```
pdadmin –a admin_id -p password cmd_filename
```

The command file would produce results like:

```
cmd> user show cwright

Login ID: cwright
LDAP DN: cn=Claude Wright,ou=Dallas,o=Tivoli,c=us
LDAP CN: Claude Wright
LDAP SN: Wright
Description:
Is SecUser: yes
Is GSO user: no
Account valid: yes
Password valid: yes
Authorization mechanism: Default:LDAP

cmd:> user show oogle
```

```
Could not perform the administration request.
Error: HPDMG0754WThe entry was not found. If ...
(status 0x14c012f2)
```

Local or other domain

Use the **pdadmin** command to authenticate your user ID and password. You must authenticate before you log in to the local domain or to a domain other than the local domain.

To authenticate and log in to your local domain, in interactive mode, enter:

```
pdadmin> login -a dlucas -p lucaspwd

pdadmin dlucas>
```

In the example, *`user_name`* logs you in as the authenticated user `dlucas` to your own local domain.

To authenticate and log in to a domain with a name that is different from the local domain, enter:

```
pdadmin> login -a dlucas -p lucaspwd -d domain_a

pdadmin dlucas@domain_a>
```

In the example, *`user_name`* logs you in as the authenticated user `dlucas`. `domain_a` is the *`domain_name`* to which you are logging on, in interactive mode.

Command option processing

Some **pdadmin** command options use specific symbols or characters.

Some **pdadmin** command options begin with a hyphen (-). For example, the following command uses the `–gsouser` option:

```
pdadmin sec_master> user import –gsouser mlucaser cn=mlucaser,o=Tivoli,c=US
```

The **pdadmin** command interprets any token beginning with a hyphen as a command option, even if the hyphen is placed within double quotation marks.

Occasionally, you might want a token that begins with a **–** to be interpreted as an argument rather than as a command option. For example, you might want to name the user `–mlucaser` or `"–mlucaser"` by entering:

```
pdadmin sec_master> user import –gsouser –mlucaser cn=mlucaser,o=tivoli,c=us
```

In this example, the first `–gsouser` option in the command is still processed. However, because the user name token begins with a hyphen, the user name would be interpreted as a command option. The command would fail because the –`mlucaser` command option does not exist.

Specify the single hyphen character to turn off the interpretation of the optional arguments, by the **pdadmin** command. Following the single hyphen character, `–mlucaser` is now interpreted as the user name.

For example:

```
pdadmin sec_master> user import -gsouser – -mlucaser cn=mlucaser,o=Tivoli,c=us
```

Options on the command line are position-independent. You can change the order so that all tokens that begin with a hyphen, which are not command options, follow the single hyphen character.

Commands by category

The **pdadmin** commands are listed here by major category.

This section lists the **pdadmin** commands by the following categories:

Access control list commands

Use **acl** commands to manage access control list (ACL) policies and extended attributes.

Table 1 on page 8 lists **acl** commands.

Table 1. Access control list (ACL) commands

Command	Description
"acl attach" on page 16	Attaches an ACL policy to a protected object. If the protected object already has an ACL attached, the ACL is replaced with a new one.
"acl create" on page 17	Creates an ACL policy in the ACL database. This command does not create ACL entries.
"acl delete" on page 18	Deletes an ACL policy from the ACL database.
"acl detach" on page 18	Detaches the current ACL policy from a protected object. This command does not delete the ACL policy from the ACL database.
"acl find" on page 19	Finds and lists all protected objects that have a specific ACL policy attached.
"acl list" on page 20	Lists the names of all defined ACLs. Also lists the extended attribute keys that are associated with a specific ACL.
"acl modify" on page 21	Modifies ACLs, their extended attributes, and associated values.
"acl show" on page 24	Lists the complete set of entries for a specific ACL policy. Also lists the values of a specific extended attribute that is associated with an ACL policy.

Action commands

The **action** commands define more authorization actions (permissions) and action groups.

Table 2 on page 8 lists **action** commands.

Table 2. Action commands

Command	Description
"action create" on page 25	Creates and adds an action to an action group.
"action delete" on page 27	Deletes an action from an action group.

Table 2. Action commands (continued)

Command	Description
"action group create" on page 28	Creates an action group.
"action group delete" on page 28	Deletes an action group.
"action group list" on page 29	Lists all action groups.
"action list" on page 30	Lists all defined actions in an action group.

Authorization rule commands

The **authzrule** commands manage authorization rules.

Table 3 on page 9 lists **authzrule** commands.

Table 3. Authorization rule commands

Command	Description
"authzrule attach" on page 31	Attaches an authorization rule to the specified protected object.
"authzrule create" on page 32	Creates an authorization rule.
"authzrule delete" on page 33	Deletes an authorization rule.
"authzrule detach" on page 34	Detaches an authorization rule from the specified protected object.
"authzrule find" on page 34	Finds and lists all the protected objects that have the specified authorization rule attached.
"authzrule list" on page 35	Lists all the registered authorization rules.
"authzrule modify" on page 36	Modifies an authorization rule.
"authzrule show" on page 37	Shows all the attributes of an authorization rule, including description, rule text, and fail reason code.

Context commands

Context commands display the context (authentication) information for the user who is running the **pdadmin** utility.

Table 4 on page 9 lists **context** commands.

Table 4. Context commands

Command	Description
"context show" on page 38	Displays the user ID and domain ID used to establish the current context.

Domain commands

Domain commands manage Security Verify Access secure domains.

Table 5 on page 10 lists **domain** commands.

Table 5. Domain commands

Command	Description
“domain create” on page 39	Creates a Security Verify Access secure domain.
“domain delete” on page 41	Deletes the specified Security Verify Access secure domain, and optionally deletes the information about the domain from the user registry.
“domain list” on page 42	Lists all the domains except for the management domain.
“domain modify” on page 42	Modifies the description of the specified domain.
“domain show” on page 43	Displays the specified attributes of the domain, including name and description.

Group commands

Group commands manage Security Verify Access groups.

A *group* is a set of Security Verify Access user accounts that have similar attributes. By using groups, you can use a group name in an access control list (ACL) instead of listing all users individually. When an LDAP-based user registry is used, group names are not case-sensitive.

Table 6 on page 10 lists **`group`** commands.

Table 6. Group commands

Command	Description
“group create” on page 46	Creates a group.
“group delete” on page 47	Deletes the specified Security Verify Access group and optionally deletes the information about the group from the user registry. ACL entries that are associated with the group are also deleted.
“group import” on page 48	Imports the information about an existing registry group to create a Security Verify Access group.
“group list” on page 49	Generates a list of all groups, by group names, whose names match the specified pattern.
“group modify” on page 50	Changes an existing group by adding a description, or adding or removing a list of members.
“group show” on page 52	Displays details about a specified group.

Login and logout commands

Login and logout commands are used to log in to, and log out of, a Security Verify Access secure domain.

Table 7 on page 10 lists **`login`** and **`logout`** commands.

Table 7. Logon commands

Command	Description
“login” on page 54	Authenticates the user to the Security Verify Access policy server as a given administrative identity in a domain.
“logout” on page 57	Discards any authentication credentials that are in effect.

Object commands

Object commands can protect objects by attaching ACLs or protected object policy (POP).

Table 8 on page 11 lists **`objects`** commands.

Table 8. Object commands

Command	Description
"object access" on page 58	Confirms whether a specified access is permitted on the named protected object.
"object create" on page 60	Creates a protected object.
"object delete" on page 62	Deletes a protected object.
"object exists" on page 63	Confirms whether a protected object is in either the policy database or in an object space that is managed by an administration service plug-in.
"object list" on page 64	Lists any objects that are grouped under the specified protected object. Also lists all the extended attributes that are associated with the specified protected object.
"object listandshow" on page 65	Lists any child objects that are grouped under the specified protected object and shows all values that are associated with each of those objects.
"object modify" on page 66	Modifies an existing object.
"object show" on page 69 "object copy" on page 59	Shows all values that are associated with a protected object. Recursively copy the policy from one protected object space to another.

Object space commands

Object space commands allow the creation of more object spaces that contain protected objects that are used by third-party applications.

Table 9 on page 11 lists **`objectspace`** commands.

Table 9. Objectspace commands

Command	Description
"objectspace create" on page 72	Creates a protected object space under which protected objects can be placed.
"objectspace delete" on page 74	Deletes an existing protected object space and all associated protected objects.
"objectspace list" on page 75	Lists all the existing protected object spaces in the policy server.

Policy commands

Policy commands manage user password and account policies.

Table 10 on page 12 lists **`policy`** commands.

Table 10. Policy commands

Command	Description
"policy get" on page 75	Displays the policy for user passwords, account rules, and conditions. Requires authentication (administrator ID and password) to use this command.
"policy set" on page 77	Sets the policy for user passwords, account rules, and conditions. Requires authentication (administrator ID and password) to use this command.

Protected object policy commands

Protected object policy commands allow the creation of a protected object policy (POP) and extended attributes for the protected object policies

Table 11 on page 12 lists **pop** commands.

Table 11. Protected object policy (POP) commands

Command	Description
"pop attach" on page 80	Attaches a protected object policy to a specified protected object.
"pop create" on page 81	Creates a protected object policy.
"pop delete" on page 82	Deletes the specified protected object policy.
"pop detach" on page 83	Detaches a protected object policy from the specified protected object.
"pop find" on page 84	Finds and lists all protected objects with protected object policies attached.
"pop list" on page 85	Lists all created protected object policies.
"pop modify" on page 85	Modifies the protected object policy.
"pop show" on page 89	Shows details about the protected object policy.

Resource and resource group commands

Resource and resource group commands manage resource-related information.

Table 12 on page 12 lists **rsrc**, **rsrccred**, and **rsrcgroup** commands.

Table 12. Resource commands

Command	Description
"rsrc create" on page 90	Creates and names a server as a resource.
"rsrc delete" on page 91	Deletes the specified single sign-on resource.
"rsrc list" on page 91	Returns a list of all the single sign-on resource names.
"rsrc show" on page 92	Displays the resource information for the named resource.
"rsrccred create" on page 93	Creates and names a resource credential.
"rsrccred delete" on page 94	Deletes only the resource credential information for an existing user.
"rsrccred list user" on page 95	Displays the names of all defined resource credentials and their type for the specified user.

Table 12. Resource commands (continued)

Command	Description
"rsrccred modify" on page 96	Changes the user ID and password resource credential information for the named resource.
"rsrccred show" on page 97	Displays the resource credential information for a specified user.
"rsrcgroup create" on page 98	Creates and names a resource group.
"rsrcgroup delete" on page 99	Deletes the named resource group, including any description information.
"rsrcgroup list" on page 100	Displays the names of all resource groups that are defined in the user registry.
"rsrcgroup modify" on page 100	Adds or removes a single sign-on resource to or from a single sign-on resource group.
"rsrcgroup show" on page 101	Displays the resource group information for the specified resource group.

Server commands

Server commands perform management tasks on Security Verify Access servers.

Table 13 on page 13 lists **`server`** and **`server task`** commands, and the **`admin show config`** command.

Table 13. Server commands

Command	Description
"admin show conf" on page 30	Displays current policy server configuration information.
"server list" on page 102	Lists all registered servers.
"server listtasks" on page 103	Retrieves the list of tasks (commands) available for this server.
"server replicate" on page 104	Notifies authorization servers to receive database updates.
"server show" on page 105	Displays the specified properties of the server.
"server task help" on page 119	Lists detailed help information about a specific **`server task`** command.
"server task stats" on page 144	Manages the gathering and reporting of statistics for Security Verify Access servers and server instances.
"server task trace" on page 151	Enables the gathering of trace information for components of installed Security Verify Access servers or server instances that support debug event tracing.

Distributed session cache commands

Distributed session cache commands perform session management tasks. These commands are available only when a Web security server and distributed session cache are configured.

Table 14 on page 14 lists **`server task`** commands.

Table 14. Server task commands

Command	Description
"server task sms replica set list" on page 137	Lists all session management replica sets in the domain.
"server task sms replica set show" on page 138	Lists all session management replica sets in the domain with the time and date each joined the realm.
"server task sms session list" on page 139	Lists all session management sessions.
"server task sms session terminate all_sessions" on page 140	Terminates all user sessions for a specific user.
"server task sms session terminate session" on page 141	Terminates a specific session.

User commands

User commands manage Security Verify Access users.

Table 15 on page 14 lists **user** commands.

Table 15. User commands

Command	Description
"user create" on page 175	Creates a Security Verify Access user account.
"user delete" on page 177	Deletes a Security Verify Access user and optionally deletes the user information from the user registry. ACL entries that are associated with the user are also deleted.
"user import" on page 177	Imports the information about an existing registry user to create a Security Verify Access user.
"user list" on page 179	Generates a list of all users whose names match the specified pattern, which is listed by user names.
"user modify" on page 180	Modifies various user account parameters.
"user show" on page 182	Displays details about a specified user.

WebSEAL commands

WebSEAL commands perform management tasks on WebSEAL servers and instances. These commands are available only when WebSEAL is installed.

Table 16 on page 14 lists **server task** commands.

Table 16. WebSEAL server task commands

Command	Description
"server task add" on page 107	Adds an application server to an existing WebSEAL junction.
"server task cache flush all" on page 109	Flushes the HTML document cache.
"server task create" on page 110	Creates a WebSEAL junction point.
"server task delete" on page 117	Deletes a WebSEAL junction point.

Table 16. WebSEAL server task commands (continued)

Command	Description
“server task dynurl update” on page 118	Reloads the dynamic URL configuration file.
“server task help” on page 119	Lists detailed help information about a specific **`server task`** command.
“server task jmt” on page 120	Clears or loads the junction mapping table data.
“server task list” on page 122	Lists all junction points on a WebSEAL server or server instance.
“server task offline” on page 123	Places the server that is at this junction in an offline operational state.
“server task online” on page 125	Places the server that is at this junction in an online operational state.
“server task refresh all_sessions” on page 126	Refreshes the credential for all sessions for a specified user.
“server task reload” on page 127	Reloads the junction mapping table from the database.
“server task remove” on page 128	Removes the specified installed WebSEAL server or server instance from a WebSEAL junction point.
“server task show” on page 130	Displays detailed information about the specified WebSEAL junction.
“server task terminate all_sessions” on page 147	Terminates all user sessions for a specific user.
“server task terminate session” on page 148	Terminates a user session by using a session ID.
“server task throttle” on page 149	Places the server that is at this junction in a throttled operational state.
“server task virtualhost add” on page 153	Adds an additional installed WebSEAL server or server instance to an existing virtual host junction.
“server task virtualhost create” on page 155	Creates a virtual host junction.
“server task virtualhost delete” on page 161	Deletes a virtual host junction.
“server task virtualhost list” on page 162	Lists all configured virtual host junctions by label name.
“server task virtualhost offline” on page 163	Places the server that is at this virtual host junction in an offline operational state.
“server task virtualhost online” on page 165	Places the server that is at this virtual host junction in an online operational state.
“server task virtualhost remove” on page 167	Removes the specified server from a virtual host junction.
“server task virtualhost show” on page 169	Displays information about the specified virtual host junction.
“server task virtualhost throttle” on page 170	Places the server that is at this virtual host junction in a throttled operational state.

Table 16. WebSEAL server task commands (continued)

Command	Description
"server task server restart" on page 172	Restarts the WebSEAL instance.
"server task server sync" on page 173	Synchronizes the configuration of the supplied WebSEAL authorization server to the current WebSEAL server.
"server task file cat" on page 174	Obtains the string content of the specified file. A flag controls whether the contents of the file are base64 encoded or not encoded. A WebSEAL configuration item defines the maximum allowable size of the file.

acl attach

Attaches an ACL policy to a protected object. If the protected object already has an ACL attached, the ACL is replaced with a new one.

Requires authentication (administrator ID and password) to use this command.

Syntax

`acl attach` *`object_name acl_name`*

Description

At most, one ACL can be attached to a given protected object. The same ACL can be attached to multiple protected objects. Ensure that you are familiar with ACL management before you use this function.

Options

acl_name

Specifies the ACL policy that is applied to the named object. The ACL policy must exist, or an error is displayed.

Examples of the ACL names are `default-root`, `test`, `default-management`, and `pubs_acl3`.

object_name

Specifies the object to which to apply the named ACL policy. The object name must exist, or an error is displayed.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example attaches the ACL policy, pubs_acl3, to the protected object, /Management:

```
pdadmin sec_master> acl attach /Management pubs_acl3
```

See also

“acl create” on page 17
“acl detach” on page 18

acl create

Creates an ACL policy in the ACL database. This command does not create ACL entries.

Requires authentication (administrator ID and password) to use this command.

Syntax

acl create *acl_name*

Options

acl_name

Specifies the name of the ACL policy that is being created. A valid ACL policy name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed. The following characters cannot be used in the name of the ACL policy:

```
! " # & ( ) * + , ; : < > = @ / \ | .
```

Examples: default-root, test, default-management, and pubs_acl3

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates an ACL policy named pubs_acl3:

```
pdadmin sec_master> acl create pubs_acl3
```

- The following example creates an ACL policy named Test-ACL:

```
pdadmin sec_master> acl create Test-ACL
```

See also

“acl attach” on page 16
“acl delete” on page 18
“acl modify” on page 21

acl delete

Deletes an ACL policy from the ACL database.

Requires authentication (administrator ID and password) to use this command.

Syntax

`acl delete` *`acl_name`*

Options

`acl_name`

Specifies the name of the ACL policy that is being deleted from the ACL database. The ACL policy must exist, or an error is displayed.

Examples: `default-root`, `test`, `default-management`, and `pubs_acl3`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the ACL policy that is named `pubs_acl3`:

```
pdadmin sec_master> acl delete pubs_acl3
```

- The following example deletes the ACL policy that is named `Test-ACL`:

```
pdadmin sec_master> acl delete Test-ACL
```

See also

"acl detach" on page 18

acl detach

Detaches the current ACL policy from a protected object. This command does not delete the ACL policy from the ACL database.

Requires authentication (administrator ID and password) to use this command.

Syntax

`acl detach` *`object_name`*

Description

Only one access control list at a time can be attached to an object. Therefore, the currently attached access control list is detached. If the object does not have an attached ACL policy, an error is displayed.

Options

object_name

Specifies the object from which the current ACL policy is being removed. The object must exist and have an ACL attached, or an error is displayed.

Examples of object names are:

- /Management/Groups/Travel
- /WebSEAL
- /Management

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example detaches the ACL from the protected object /Management:

```
pdadmin sec_master> acl detach /Management
```

See also

"acl attach" on page 16
"acl delete" on page 18
"acl modify" on page 21

acl find

Returns a list of protected objects, which have the specified ACL attached.

Requires authentication (administrator ID and password) to use this command.

Syntax

acl find *acl_name*

Description

A user must have the browse (**b**) and view (**v**) permissions for the object to be listed when the **pdadmin object show** command is issued. Otherwise, an error is returned:

```
The user is not authorized to view one or more protected objects where the
requested acl is attached.
```

Options

acl_name

Specifies the name of the ACL policy that you want to find. The ACL policy must exist, or an error is displayed.

Examples: default-root, test, default-management, and pubs_acl3

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists the protected object that has the `default-config` ACL attached:

  ```
  pdadmin sec_master> acl find default-config
  ```

  Provides output like:

  ```
  /Management/Config
  ```

- The following example lists the protected objects that have the user-defined ACL, `_WebAppServer_deployedResources_CosNamingDelete_admin_ACL`, attached:

  ```
  pdadmin sec_master> acl find
  _WebAppServer_deployedResources_CosNamingDelete_admin_ACL
  ```

  Provides output like:

  ```
   /WebAppServer/deployedResources/CosNamingDelete/admin
  ```

See also

"acl list" on page 20
"acl show" on page 24

acl list

Lists the names of all defined access control lists. Alternatively, lists the extended attribute keys that are associated with a specific ACL.

Requires authentication (administrator ID and password) to use this command.

Syntax

acl list

acl list [*acl_name* attribute]

acl list [*pattern max-return*]

Options

***acl_name* attribute**

Specifies the ACL policy for which to list the attributes. The ACL policy must exist, or an error is displayed. (Optional)

Examples: `default-root`, `test`, `default-management` and `pubs_acl3`.

pattern max-return

Specifies the pattern and the maximum number of access control lists to be returned. (Optional)

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists ACL policies:

```
pdadmin sec_master> acl list
```

The output is like:

```
default-webseal
default-root
test
default-replica
default-management
```

See also

“acl find” on page 19
“acl show” on page 24

acl modify

Modifies access control list (ACL) policies.

Requires authentication (administrator ID and password) to use this command.

Syntax

acl modify *acl_name* delete attribute *attribute_name* [*attribute_value*]

acl modify *acl_name* description *description*

acl modify *acl_name* remove any-other

acl modify *acl_name* remove group *group_name*

acl modify *acl_name* remove unauthenticated

acl modify *acl_name* remove user *user_name*

acl modify *acl_name* set any-other [*permissions*]

acl modify *acl_name* set attribute *attribute_name attribute_value*

acl modify *acl_name* set description *description*

acl modify *acl_name* set group *group_name* [*permissions*]

acl modify *acl_name* set unauthenticated [*permissions*]

acl modify *acl_name* set user *user_name* [*permissions*]

Options

acl_name

Specifies the ACL policy that you want to be modified. The ACL policy must exist, or an error is displayed.

Examples: `default-root`, `test`, `default-management`, and `pubs_acl3`

delete attribute *attribute_name* [*attribute_value*]

Deletes the specified extended attribute name and value from the specified ACL. The attribute must exist, or an error is displayed.

The *attribute_value* deletes the specified value from the specified extended attribute key in the specified ACL. (Optional)

Examples of extended attribute names and values:

```
Dept_No 445
Employee_Name "Diana Lucas"
```

description *description*

Sets or modifies the description for the specified ACL. This option is equivalent to the **acl modify set description** command. Use the **acl modify description** command instead of the **acl modify set description** command.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are allowed.

If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: `"Department number of employee"`

permissions

Security Verify Access uses a set of default actions (known as *primary* action tasks and permissions) that cover a wide range of operations. You can also create your own action tasks and permissions.

A complete list of primary action tasks and their associated permissions includes:

```
T Traverse Base
c Control Base
g Delegation Base
m Modify Generic
d Delete Generic
b Browse Base
s Server Admin Generic
v View Generic
a Attach Base
B Bypass POP Base
t Trace Base
r Read WebSEAL
x Execute WebSEAL
l List Directory WebSEAL
N Create Base
W Password Base
A Add Base
R Bypass AuthzRule Base
```

For more information on actions, see Action groups and actions. For a description of default permissions, see Default permissions in the primary action group.

remove any-other

Removes the ACL entry for the **any-other** user category from the specified ACL.

remove group *group_name*

Removes the ACL entry for the specified group from the specified ACL. The group must exist, or an error is displayed.

Examples of group names are `Credit`, `Sales`, and `Test-group`.

remove unauthenticated
Removes the ACL entry for the **unauthenticated** user category from the specified ACL.

remove user *user_name*
Removes the ACL entry for the specified user from the specified ACL. The user must exist, or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

set any-other [*permissions*]
Sets or modifies the ACL entry for the **any-other** user category in the ACL. Valid actions, or *permissions*, are represented by single alphabetic ASCII characters (a-z, A-Z).

set attribute *attribute_name attribute_value*
Sets the extended attribute value for the specified extended attribute key in the specified ACL. The attribute must exist, or an error is displayed. If the attribute exists, the attribute value is added as an additional value if the same value does not exist for this attribute. If the same value exists for this attribute, it does not get added again (duplicate values are not allowed), and no error is returned.

The optional *`attribute_value`* sets the specified value from the specified extended attribute key in the specified ACL.

Examples of extended attribute names and values:

```
Dept_No 445
Employee_name "Diana Lucas"
```

set description *description*
Sets or modifies the description for the specified ACL. This option is equivalent to the **acl modify description** command. Use the **acl modify description** command instead of the **acl modify set description** command.

set group *group_name* [*permissions*]
Sets or modifies the ACL entry for the specified group in the specified ACL. The group must exist, or an error is displayed.

Examples of group names are `Credit`, `Sales`, and `Test-group`.

Security Verify Access uses a set of default actions that cover a wide range of operations. Valid actions, or permissions, are represented by single alphabetic ASCII characters (a-z, A-Z). See `set any-other` [*permissions*] for the list of possible permissions.

set unauthenticated [*permissions*]
Sets or modifies the ACL entry for the **unauthenticated** user category in the specified ACL.

Security Verify Access uses a set of default actions that cover a wide range of operations. Valid actions, or permissions, are represented by single alphabetic ASCII characters (a-z, A-Z). See `set any-other` [*permissions*] for examples of permissions.

set user *user_name* [*permissions*]
Sets permissions that the user is permitted to perform. The user must exist or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Security Verify Access uses a set of default actions that cover a wide range of operations. Valid actions, or permissions, are represented by single alphabetic ASCII characters (a-z, A-Z). See `set any-other` [*permissions*] for examples of permissions.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example sets the `any-other` user entry in the pubs ACL to have `r`, the Read (WebSEAL) permission:

  ```
  pdadmin sec_master> acl modify pubs set any-other r
  ```

- The following example sets the `sales` group entry in the `pubs` ACL to have the `Tr` permissions, which are the Traverse and Read (Base) permissions:

  ```
  pdadmin sec_master> acl modify pubs set group sales Tr
  ```

- The following example sets the `unauthenticated` user entry in the `docs` ACL to have the `r` permission, which is the Read (WebSEAL) permission:

  ```
  pdadmin sec_master> acl modify docs set unauthenticated r
  ```

- The following example sets the `peter` user entry in the `pubs` ACL to have the `Tr` permissions, which are the Traverse (Base) and Read (WebSEAL) permissions:

  ```
  pdadmin sec_master> acl modify pubs set user peter Tr
  ```

- The following example sets the `kathy` user entry in the `test` ACL to have `Tbr` permissions, which are the Traverse (Base), Browse (Base) and Read (WebSEAL) permissions. It also sets custom permissions PS for the existing `test-group` action group. It then displays the results.

  ```
  pdadmin sec_master> acl modify test set user kathy Tbr[test-group]PS

  pdadmin sec_master> acl show test

  ACL Name: test
  Description:
  Entries:
  User sec_master TcmdbsvaBl
  Group ivmgrd-servers Tl
  Any-other r
  User kathy Tbr[test-group]PS
  ```

- The following example sets the `kathy` user entry in the `test` ACL to have `Tbr` permissions, which are the Traverse (Base), Browse (Base), and Read (WebSEAL) permissions. It then displays the results.

  ```
  pdadmin sec_master> acl modify test set user kathy Tbr

  pdadmin sec_master> acl show test

  ACL Name: test
  Description:
  Entries:
  User sec_master TcmdbsvaBl
  Group ivmgrd-servers Tl
  Any-other r
  User kathy Tbr
  ```

See also

"acl attach" on page 16
"acl create" on page 17

acl show

Lists the complete set of entries for a specific access control list (ACL) policy. Alternatively, lists the values of a specific extended attribute that is associated with an ACL policy.

Requires authentication (administrator ID and password) to use this command.

Syntax

acl show *acl_name* [attribute *attribute_name*]

Options

acl_name

Specifies the name of the ACL policy for which the extended attribute values are displayed. The ACL policy must exist, or an error is displayed.

Examples of ACL names are `default-root`, `test`, `default-management`, and `pubs_acl3`.

attribute ***attribute_name***

Specifies the name of the extended attribute whose values are displayed. (Optional) The system handles this command as follows:

- If the ACL either has an attribute or had an attribute in the past, no error is displayed.
- If the ACL never had an attribute, then an error is displayed.

Examples of extended attribute names are `Dept_No` and `Employee_Name`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example shows details about ACL `test-acl`:

```
pdadmin sec_master> acl show test-acl

ACL Name: test-acl
Description:
Entries:
User sec_master Tcmdbva
Group ivmgrd-servers Tl
Any-other r
```

See also

“acl find” on page 19
“acl list” on page 20

action create

Creates and adds an action (permission) to an action group.

Requires authentication (administrator ID and password) to use this command.

Syntax

action create *action_name action_label action_type* [*action_group_name*]

Description

Action codes (permissions) consist of one alphabetic character (a-z or A-Z) and are case-sensitive. Each action code can be used only once within an action group. Ensure that you do not attempt to redefine the default action codes when you add custom codes to the primary group.

Options

action_group_name

Specifies the name of the action group to which the action code is to be added. If no action group is specified, the action is added to the `primary` action group. Supports a maximum of 32 action groups. Examples of action group names are `primary` and `test-group`. (Optional)

action_label

Specifies the label or description for the action. Each default permission is displayed with a label that describes the operation that it governs. In addition, the ACLs are grouped in one of the following ways, according to their use:

- In a particular part of the `objectspace`, such as, `WebSEAL`.
- Across the entire `objectspace`, such as, `Base`, `Generic`.

For example, `time` is the action label in the following example:

```
k time Ext-Authzn
```

A valid action label is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Examples of action labels: `time`, `Generic`, `Base`, and `WebSEAL`

action_name

Specifies the new single-character permission that is being created, which can be specified by using any case.

Security Verify Access uses a set of default actions that cover a wide range of operations. Valid actions, or *permissions*, are represented by single alphabetic ASCII characters (a-z, A-Z).

For example, k is the action name in the following example:

```
k time Ext-Authzn
```

action_type

Specifies the organizational category for this action within a specified action group. The action type can be a description of the action, such as what application the action is specific to. The action type is application-specific and typically refers to:

- The application that defined the action, such as, `WebSEAL`.
- The function that uses the action, such as, `Ext-Authzn`, for extended authorization checks.

A valid action type is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

For example, `Ext-Authzn` is the action type in the following example:

```
k time Ext-Authzn
```

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates an action code named k with an action label of `time` and an action type of `Ext-Authzn` within the `primary` action group:

```
pdadmin sec_master> action create k time Ext-Authzn
```

- The following example creates a customized action named P and an action label of `Test-Action` with an action type of `Special` within the `test-group` action group:

```
pdadmin sec_master> action create P Test-Action Special test-group
```

See also

"action delete" on page 27

action delete

Deletes an action (permission) from an action group.

Requires authentication (administrator ID and password) to use this command.

Syntax

action delete *action_name* [*action_group_name*]

Options

action_group_name

Specifies the name of the action group from which the specified action must be deleted. Examples of action group names are `primary` and `test-group`. (Optional)

action_name

Specifies the name of the action to be deleted. The action code must exist, or an error is displayed.

Security Verify Access uses a set of default actions that cover a wide range of operations. Valid actions, or *permissions*, are represented by single alphabetic ASCII characters (a-z, A-Z). For example, k is the action name in the following example:

```
k time Ext-Authzn
```

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes action k from the `primary` action group:

```
pdadmin sec_master> action delete k
```

- The following example deletes the action z from the agz action group:

```
pdadmin sec_master> action delete z agz
```

See also

"action create" on page 25

action group create

Creates an action group.

Requires authentication (administrator ID and password) to use this command.

Syntax

action group create *action_group_name*

Options

action_group_name

Specifies the name of the action group to create. Supports a maximum of 32 action groups. The action group must not exist, or an error is displayed.

A valid action group name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Examples of action group names are `primary` and `test-group`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example creates the `test` action group:

```
pdadmin sec_master> action group create test
```

action group delete

Deletes an action group.

Requires authentication (administrator ID and password) to use this command.

Syntax

action group delete *action_group_name*

Options

action_group_name

Specifies the name of the action group to delete. All the actions that belong to the specified group are also deleted. The action group must exist, or an error is displayed. Examples of action group names are `primary` and `test-group`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example deletes the `test` action group:

```
pdadmin sec_master> action group delete test
```

action group list

Lists all action groups.

Requires authentication (administrator ID and password) to use this command.

Syntax

action group list

Description

The **action group list** command lists all the defined names of action groups

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists the names of all defined action groups:

```
pdadmin sec_master> action group list
               primary
               test-group
```

action list

Lists all actions (permissions) in an action group.

Requires authentication (administrator ID and password) to use this command.

Syntax

action list [*action_group_name*]

Options

action_group_name

Specifies the name of the action group for which all actions are displayed. If this option is not specified, actions that are defined in the `primary` action group are listed. The action group must exist, or an error is displayed.

Examples of action group names are `primary` and `test-group`. (Optional)

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example displays all existing actions in the `primary` action group:

```
pdadmin sec_master> action list
```

```
TTraverseBase
cControlBase
gDelegationBase
mModifyGeneric
dDeleteGeneric
bBrowseBase
sServer AdminGeneric
vViewGeneric
aAttachBase
BBypass POPBase
tTraceBase
rReadWebSEAL
xExecuteWebSEAL
lList DirectoryWebSEAL
NCreateBase
WPasswordBase
AAddBase
RBypass AuthzRuleBase
```

admin show conf

Displays the current policy server configuration information, such as the type of registry or whether global sign-on is enabled.

Requires authentication (administrator ID and password) to use this command.

Syntax

admin show conf

Options

None.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example displays the current server configuration information:

```
pdadmin sec_master> admin show conf

LDAP: yes
secAuthority
GSO: yes
```

authzrule attach

Attaches an authorization rule to the specified protected object.

Requires authentication (administrator ID and password) to use this command.

Syntax

`authzrule attach` *`protobjid`* *`ruleid`*

Description

At most, one rule can be attached to a given protected object. If the object already has a rule that is attached to it, the specified rule replaces the existing one. The same rule can be attached to multiple protected objects. Ensure that the protected object exists in the protected object space before you attach a rule.

Options

protobjid
Specifies the fully qualified name of the protected object to which the authorization rule is attached. The object must exist, or an error is displayed.

ruleid
Specifies the name of the authorization rule to attach. The rule must exist, or an error is displayed.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example attaches the `r1` rule to the `/Test-Space/folder1` protected object named:

```
pdadmin sec_master> authzrule attach /Test-Space/folder1 r1
```

See also

“authzrule create” on page 32
“authzrule detach” on page 34

authzrule create

Creates an authorization rule.

Requires authentication (administrator ID and password) to use this command.

Syntax

`authzrule create` *rule_id* *rule_text* [–desc *description*] [–failreason *fail_reason*]

Description

You can attach an authorization rule to a protected object. To authorize access to the protected object, the user credential and application context attributes are compared against the rule.

Note: Quotation marks within an authorization rule must be escaped by using the backward slash (\) character.

Options

–desc *description*
Specifies the description of the authorization rule. (Optional)

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: `"time-of-day rule for engineering object space"`

–failreason *fail_reason*
Specifies the message that is returned if the rule denies access to a protected object. Consider that the authorization is denied as a result of the evaluation of this rule. However, other authorization checks succeed. In this case, the reason code is returned to the application that makes the authorization check. (Optional)

rule_id
Specifies the name of the authorization rule to create.

A valid authorization rule is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed. The following characters cannot be used in the name of an authorization rule:

```
! " # & ( ) * + , ; : < > = @ / \ | .
```

rule_text
Specifies the rule policy that is used to evaluate the rule in XSL format. The rule must be enclosed in double quotation mark (") character. If the rule specifies a double quotation mark as part of the rule text, precede the double quotation mark with a backward slash (\) character. Doing so instructs the system to ignore the double quotation mark.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

When providing rule text with the **pdadmin** utility, enclose the rule text in double quotation marks ("). Double quotation marks embedded within the rule text must be escaped with a backward slash (\) so that they are ignored by the **pdadmin** utility. The XSL processor treats single and double quotation marks equally for defining text strings. They can be used interchangeably, but they must always be paired appropriately. For example:

```
pdadmin sec_master> authzrule create testrule1
"<xsl:if test='some_piece_of_ADI =\"any string\"'>!TRUE!</xsl:if>"
```

See also

"authzrule delete" on page 33

authzrule delete

Deletes an authorization rule.

Requires authentication (administrator ID and password) to use this command.

Syntax

authzrule delete *rule_id*

Options

rule_id

Specifies the name of the authorization rule to delete. The authorization rule must exist, or an error is displayed.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

The following example deletes the `eng-test` rule:

```
pdadmin sec_master> authzrule delete eng-test
```

The following example deletes the `myRule` rule:

```
pdadmin sec_master> authzrule delete myRule
```

See also

"authzrule create" on page 32
"authzrule detach" on page 34

authzrule detach

Detaches an authorization rule from the specified protected object.

Requires authentication (administrator ID and password) to use this command.

Syntax

`authzrule detach` *`protobjid`*

Options

`protobjid`
Specifies the name of the protected object from which the authorization rule is detached. The object must exist and have an authorization rule that is attached, or an error is displayed.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example detaches a rule from the `/WebSEAL/tivoli.com/w3junction/index.html` protected object:

```
pdadmin sec_master> authzrule detach /WebSEAL/tivoli.com/w3junction/index.html
```

See also

"authzrule attach" on page 31
"authzrule delete" on page 33

authzrule find

Finds and lists all protected objects that have the specified authorization rule attached.

Requires authentication (administrator ID and password) to use this command.

Syntax

`authzrule find` *`rule_id`*

Description

A user must have the browse (**b**) and view (**v**) permissions for the object to be listed when the **pdadmin object show** command is issued. Otherwise, an error is returned:

```
The user is not authorized to view one or more protected objects where the
requested authzrule is attached.
```

Options

rule_id

Specifies the name of the authorization rule to find. The authorization rule must exist, or an error is displayed.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example finds protected objects that are attached to the `r2` rule:

```
pdadmin sec_master> authzrule find r2

/Marketing/Folder1
```

See also

“authzrule list” on page 35

authzrule list

Lists all the authorization rules.

Requires authentication (administrator ID and password) to use this command.

Syntax

authzrule list

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists authorization rules:

```
pdadmin sec_master> authzrule list

r1
r2
r3
r4
```

See also

"authzrule find" on page 34

authzrule modify

Changes an authorization rule.

Requires authentication (administrator ID and password) to use this command.

Syntax

authzrule modify *rule_id* ruletext *rule_text*

authzrule modify *rule_id* description *description*

authzrule modify *rule_id* failreason *fail_reason*

Options

description *description*
Specifies the new description of the rule.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are allowed. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: `"time-of-day access"`

failreason *fail_reason*
Specifies the fail reason code. Consider that authorization is denied as a result of the evaluation of this rule. However, other authorization checks succeed. In this case, the reason code is returned to the application that makes the authorization check. You can specify an empty string ("") to clear an existing fail reason.

rule_id
Specifies the name of the authorization rule to change. The authorization rule must exist, or an error is displayed.

ruletext *rule_text*
Specifies the new rule text in XSL format.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example changes the description of a rule named r2:

```
pdadmin sec_master> authzrule modify r2 description "time-of-day access"
```

See also

"authzrule attach" on page 31
"authzrule create" on page 32

authzrule show

Shows all the attributes of an authorization rule, including description, rule text, and fail reason code.

Requires authentication (administrator ID and password) to use this command.

Syntax

authzrule show *rule_id*

Options

rule_id
Specifies the name of the authorization rule to show. The rule must exist, or an error is displayed.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example shows attributes for a rule named r2:

```
pdadmin sec_master> authzrule show r2
```

The output is like:

```
Authorization Rule Name: r2
Description: time-of-day access
Rule Text: <xsl:if test="/XMLADI/session[contains(status,'login')]">
<xsl:for-each select="/XMLADI/userid/level">
<xsl:if test=". = 'administrator'">
<xsl:choose>
<xsl:when test="../paid = 'in-full'">
!TRUE!
</xsl:when>
<xsl:when test="../paid = 'partial'">
!FALSE!
</xsl:when>
<xsl:when test="../paid = 'introductory'">
!TRUE!
</xsl:when>
<xsl:otherwise>
!FALSE!
</xsl:otherwise>
</xsl:choose>
</xsl:if>
</xsl:for-each>
```

```
</xsl:if>
Fail Reason:Error when creating R2
```

See also

"authzrule find" on page 34
"authzrule list" on page 35

context show

Displays the user ID and domain ID used to establish the current authentication context. Also, specifies whether the domain is the management domain or a domain other than the management domain.

This command does not require a login or authentication to use.

Syntax

`context show`

Options

None.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example shows that no login and no authentication are being performed:

  ```
  c:\> pdadmin

  context show
  ```

  The output is like:

  ```
  No login information
  ```

- The following example shows local authentication before the **`context show`** command is issued:

  ```
  c:\> pdadmin -l
  pdadmin local> context show
  ```

  The output is like:

  ```
  The user is logged in to the local system
  ```

- The following example shows local authentication, like the previous example, except the command in issued interactively:

  ```
  pdadmin sec_master> login -l

  pdadmin local> context show
  ```

  The output is like:

```
The user is logged in to the local system
```

- The following example shows authentication context information for a user who is logged in to the management domain (non-local authentication).

  ```
  c:\> pdadmin -a sec_master -p mypwd -m

  pdadmin sec_master> context show
  ```

  The output is like:

  ```
  User: sec_master
  Domain: Default
  The user is logged into the management domain
  ```

- The following example shows authentication context information for the `testdomain_admin` administrator who logs in interactively to a domain (`testdomain`) other than the management domain:

  ```
  pdadmin> login -a testdomain_admin -p testpwd -d testdomain

  pdadmin testdomain_admin@testdomain_admin> context show
  ```

  The output is like:

  ```
  User: testdomain_admin
  Domain: testdomain
  The user is not logged in to the management domain
  ```

See also

"domain show" on page 43
"user show" on page 182
"login" on page 54
"logout" on page 57

domain create

Creates a domain, including an administrator ID and password to log in to the specified domain. You must log in to the management domain as an administrator to perform this command.

Requires authentication (administrator ID and password) to use this command.

This command applies to LDAP registries only.

Syntax

domain create *domain domain_admin_id domain_admin_password* [–desc *description*]

Description

An initial domain is created when the policy server is configured. This domain, called the *management domain*, is the default domain in which Security Verify Access enforces security policies for authentication, authorization, and access control. You must log in to the management domain to create more policy domains.

When you create a domain, you must specify an administrative ID and password for the domain. The administrator of the management domain later assigns the new ID and password. The new credentials are assigned to the administrator responsible for handling policy management tasks for the specific domain. The administrator of the domain is responsible for updating the security policy for that particular domain if:

- Users change.
- Groups change.

- Resources change.

This domain administrator can also delegate administration tasks to others within that specific domain. For more information about managing domains, see the Administering topics in the IBM Knowledge Center.

Options

–desc *description*
Specifies an optional description for the domain. A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description. Examples of description: `"accounting area"`. (Optional)

domain
Specifies the name of the domain to be created. Characteristics of the name are:

- Limited to 64 characters in length.
- Case sensitive.
- Can contain a-z, A–Z, 0–9, hyphen (-), underscore (_), period (.), at sign (@), or ampersand(&).
- Can contain any character from a double-byte character set.

The underlying user registry might also restrict certain characters. Some registries are not case-sensitive.

domain_admin_id
Specifies an administrator ID, which is created in the specified domain.

domain_admin_password
Specifies the password for the *domain_admin_id* user.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates a domain named `Marketing`, a domain administrator ID `Admin1`, and an initial password to log in to the domain:

```
pdadmin sec_master> domain create Marketing Admin1 password
```

- The following example creates a domain named `Finance`, a domain administrator ID `Admin2`, a password, and a domain description:

```
pdadmin sec_master> domain create Finance Admin2 password
-desc "accounting area"
```

See also

“domain delete” on page 41
“domain list” on page 42
“domain modify” on page 42
“domain show” on page 43

domain delete

Deletes a domain, excluding the management domain. Optionally deletes the user and group information of the domain, from the user registry. To perform this command, you must log in to the management domain as an administrator.

Requires authentication (administrator ID and password) to use this command.

This command applies to LDAP registries only.

Syntax

`domain delete` *`domain`* `[–registry]`

Description

A domain can be deleted within the management domain only by an administrator with the appropriate privileges.

Options

`-registry`

Specifies that the information of the domain, including user and group data, be deleted from the user registry. (Optional) If this option is not selected, user and group data for the specified domain:

- Remains in the registry.
- Can be used again if the domain is re-created.

`domain`

Specifies the name of the domain to be deleted. The domain must exist, or an error is displayed.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes a domain named `Marketing`:

  ```
  pdadmin sec_master> domain delete Marketing
  ```

- The following example deletes a domain named `Finance` and removes any user and group information in the user registry:

  ```
  pdadmin sec_master> domain delete Finance -registry
  ```

See also

“domain create” on page 39
“domain list” on page 42
“domain modify” on page 42
“domain show” on page 43

domain list

Lists all domains, excluding the management domain. You must log in to the management domain as an administrator to perform this command.

Requires authentication (administrator ID and password) to use this command.

This command applies to LDAP registries only.

Syntax

`domain list`

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists existing domains other than the management domain (`Default`):

```
pdadmin sec_master> domain list
```

The output is like:

```
Marketing
Finance
Advertising
Receiving
```

See also

“domain create” on page 39
“domain delete” on page 41
“domain modify” on page 42
“domain show” on page 43

domain modify

Changes the description of a domain. You must log in to the management domain as an administrator to perform this command.

Requires authentication (administrator ID and password) to use this command.

This command applies to LDAP registries only.

Syntax

`domain modify` *`domain`* `description` *`description`*

Options

description ***description***

Specifies a new description for the domain.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are allowed. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: `"marketing and advertising areas"`

domain

Specifies the name of the domain to modify.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example changes the description that is specified for the `Marketing` domain:

```
pdadmin sec_master> domain modify Marketing description "marketing and
advertising areas"
```

See also

“domain create” on page 39
“domain delete” on page 41
“domain list” on page 42
“domain show” on page 43

domain show

Displays the properties of a domain. You must log in to the management domain as an administrator to perform this command.

Requires authentication (administrator ID and password) to use this command.

This command applies to LDAP registries only.

Syntax

domain show *domain*

Options

domain

Specifies the name of the domain for which to display properties. The domain must exist, or an error is displayed.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example displays properties for the `Marketing` domain:

```
pdadmin sec_master> domain show Marketing
```

The output is like:

```
Domain Name: Marketing
Description: marketing and advertising areas
```

See also

“domain create” on page 39
“domain delete” on page 41
“domain list” on page 42
“domain modify” on page 42

errtext

Displays the error message of a specific error number.

For detailed information about messages, see "Error messages" in the IBM Knowledge Center.

This command does not require a login or authentication to use.

Syntax

errtext *error_number*

Description

The message ID is also displayed (for example, HPDMS4047E) The message ID consists of 10 alphanumeric characters that uniquely identify the message. The message ID is composed of the following pieces:

- A three-character product identifier (for example, HPD indicates that this message is for Security Verify Access base or Web Portal Manager)
- A two-character component or subsystem identifier
- A four-digit message number
- A one-character type code indicates the severity of the message (I for informational, W for warning, and E for error)

Options

error_number

Specifies the number, in either decimal or hexadecimal, of the error for which to generate the error text.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the error message that is associated with a specific hexadecimal number:

  ```
  pdadmin sec_master> errtext 0x14c52fcf
  ```

  The output is like:

  ```
  HPDMS4047E:Non-local authentication (login) is required to perform
  this operation (status 0x14c52fcf)
  ```

- The following example displays the error message that is associated with a specific decimal number:

  ```
  pdadmin> errtext 268808652
  ```

  The output is like:

  ```
  HPDAC0460E The protected object space specified already exists in the
  authorization policy database (status 0x1005b1cc)
  ```

exit or quit

Exits from the **pdadmin** utility interactive command-line mode.

This command does not require a login or authentication to use.

Syntax

exit

quit

Options

None.

Examples

- The following example displays how to exit the **pdadmin** utility:

  ```
  pdadmin> exit
  ```

- The following example displays how to quit the **pdadmin** utility:

  ```
  pdadmin> quit
  ```

See also

“login” on page 54
“logout” on page 57
“context show” on page 38

group create

Creates a Security Verify Access group.

Requires authentication of administrator ID and password to use this command.

Groups that are created in the Active Directory Lightweight Directory Service (AD LDS) user registry must be created in the same AD LDS partition where the Security Verify Access Management Domain information is stored.

Syntax

`group create` *group_name dn cn* [*group_container*]

Options

cn
Specifies the common name that is assigned to the group that is being created. For example, `cwright`.

dn
Specifies the registry identifier that is assigned to the group that is being created.

The format for a distinguished name is like:

```
cn=credit,ou=Austin,o=Tivoli,c=US
```

group_container
Specifies the group container object that is assigned to the group that is being created. If this option is not specified, the group by default is placed in the object space under `/Management/Groups`. (Optional)

Examples of group containers are `Credit` and `Sales_Teams`.

group_name
Specifies the name of the group that is being created. This name must be unique within the domain.

A valid group name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Examples of group names are `Credit`, `Sales`, and `Test-group`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates a group named credit1 with a common name of credit01 within the Credit group container object:

```
pdadmin sec_master> group create credit1 "cn=credit01,o=Tivoli,c=US"
credit01 Credit
```

- The following example creates a group named salesteam with a common name of sales within the Sales_Teams group container:

```
pdadmin sec_master> group create salesteam "cn=sales,o=tivoli,c=us"
sales Sales_Teams
```

See also

"group delete" on page 47
"group import" on page 48

group delete

Deletes the specified Security Verify Access group. Optionally deletes the information of the group, from the user registry. ACL entries that are associated with the group are also deleted.

Requires authentication (administrator ID and password) to use this command.

Syntax

group delete [–registry] *group_name*

Options

-registry
Deletes the entire group object from the user registry. (Optional)

group_name
Specifies the name of the Security Verify Access group to be deleted. The group must exist, or an error is displayed.

Examples of group names are Credit, Sales, and Test-group.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the existing engineering group:

```
pdadmin sec_master> group delete engineering
```

- The following example deletes the group object from the user registry and also deletes the existing Test-group group:

```
pdadmin sec_master> group delete -registry Test-group
```

See also

“group create” on page 46
“group import” on page 48

group import

Creates a Security Verify Access group by importing group data that exists in the user registry.

You can import an Active Directory dynamic group under this condition:

The name of the Security Verify Access group (excluding the @*domain* suffix) is the same as the common name (CN) of the Active Directory dynamic group.

If Active Directory Lightweight Directory Service (AD LDS) is the user registry, import groups from the AD LDS partition where the Security Verify Access management domain information is stored.

Requires authentication (administrator ID and password) to use this command.

Syntax

`group import` *`group_name`* *dn* [*`group_container`*]

Options

dn

Specifies the registry identifier of the group to import. The distinguished name must exist, or an error is displayed. The format for a distinguished name is like `"cn=engineering,ou=Austin,o=Tivoli,c=us"`

`group_container`

Specifies the group container object that is assigned to the group that is being created. By default, the group is placed in the object space under `/Management/Groups`. If the container object does not currently exist, it is automatically created. (Optional)

`group_name`

Specifies the name of the group to create. A valid group name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed. Examples of group names are `Credit`, `Sales`, and `Test-group`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates a Security Verify Access group by importing a group that exists in the user registry:

```
pdadmin sec_master> group import engineering "cn=engineering,o=Tivoli,c=US"
```

- This example:
  - Creates a Security Verify Access group named `sales`.

- Places the `sales` group in the `Sales2003` group container object by importing a group that exists in the user registry.

```
pdadmin sec_master> group import sales "cn=sales,o=tivoli,c=us" Sales2003
```

- This example creates a group named `dyngroup1` by importing the group from an Active Directory dynamic group with the following characteristics:

  cn
  `dyngroup1`

  distinguishedName

  ```
  cn=dyngroup1,
  cn=AzGroupObjectContainer-myAuthorizationStore,
  cn=myAuthorizationStore,
  cn=ProgramData,
  dc=domain,
  dc=com
  ```

  ```
  pdadmin sec_master> group import dyngroup1 "cn=dyngroup1,
  cn=AzGroupObjectContainer-myAuthorizationStore,
  cn=myAuthorizationStore,cn=ProgramData,
  dc=domain,dc=com"
  ```

  If Security Verify Access is configured in an environment that uses multiple Active Directory domains, enter the following command to create the same group:

  ```
  pdadmin sec_master> group import dyngroup1@domain.com "cn=dyngroup1,
  cn=AzGroupObjectContainer-myAuthorizationStore,
  cn=myAuthorizationStore,cn=ProgramData,
  dc=domain,dc=com"
  ```

See also

"group create" on page 46

group list

Generates a list of all groups, by group names, whose names match the specified pattern.

Requires authentication (administrator ID and password) to use this command.

Syntax

`group list` *`pattern`* *`max_return`*

`group list-dn` *`pattern`* *`max_return`*

Options

`list` ***`pattern max_return`***

Specifies the pattern for the group name for which to be searched. The pattern can include a mixture of wildcard and string constants, and is not case-sensitive (for example, `*austin*`).

The *max_return* option specifies the limit of how many entries must be returned for a single request; for example, 2. The number that is returned is also governed by the server configuration. The configuration specifies the maximum number of results that can be returned as part of a search operation. The actual maximum returned entries is the minimum of *max_return* and the configured value on the server.

list-dn ***pattern max_return***

Lists user registry identifiers whose user registry common name attribute matches the pattern specified. The returned list contains groups, which are defined in the user registry. The groups might not necessarily be Security Verify Access groups. You can import groups that are not Security Verify Access groups into Security Verify Access by using the **group import** command.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists 3 groups that match the specified pattern of a group name that contains the letter a:

  ```
  pdadmin sec_master> group list *a* 3
  ```

  The output is like:

  ```
  Sales
  Marketing
  Alex
  ```

- The following example lists 2 groups that match the specified pattern of a distinguished name that contains the letter `t`:

  ```
  pdadmin sec_master> group list-dn *t* 2
  ```

  The output is like:

  ```
  cn=credit,ou=Austin,o=Tivoli,c=US Sales
  cn=marketing,ou=Boston,o=Austin Sale,c=US Marketing
  ```

See also

"group show" on page 52

group modify

Changes an existing group by adding or changing a group description, adding members to the group, or removing members from the group.

Requires authentication (administrator ID and password) to use this command.

Syntax

group modify *group_name* add *user*

group modify *group_name* add (*user_1 user_2* [... *user_n*])

group modify *group_name* description *description*

group modify *group_name* remove *user*

group modify *group_name* remove (*user_1 user_2* [... *user_n*])

Options

add *user* or add (*user_1 user_2* [... *user_n*])

Adds a user or a list of users to the group. For a single user, do not include the user name in parentheses. For multiple users, the format of the user list is a parenthesized list of user names, which are separated by spaces.

The following list shows examples of user names:

- dlucas
- "Bob Smith"
- (dlucas "Mary Jones" mlucaser)

description *description*

Changes the description for the specified group. A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are allowed. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description. For example, you can specify "Credit, Dept HCUS" as the description.

group_name

Specifies the name of the group. The group must exist, or an error is displayed.

Examples of group names are Credit, Sales, and Test-group.

remove *user* or remove (*user_1 user_2* [... *user_n*])

Removes a user or a list of users from the group. For a single user, do not include the user name in parentheses. For multiple users, the format of the user list is a parenthesized list of user names, which are separated by spaces. The following list shows examples of user names:

- dlucas
- "Bob Smith"
- (dlucas "Mary Jones" mlucaser)

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example adds a user dlucas to the engineering group:

```
pdadmin sec_master> group modify engineering add dlucas
```

- The following example adds three new users to the engineering group:

```
pdadmin sec_master> group modify engineering add ("Mary Jones" dsmith mlucaser)
```

- The following example deletes three existing users from the engineering group:

```
pdadmin sec_master> group modify engineering remove ("Mary Jones"
dlucas mlucaser)
```

- The following example changes the description of the credit group:

```
pdadmin sec_master> group modify credit description "Credit, Dept HCUS"
```

See also

“group create” on page 46
“group import” on page 48

group show

Shows the properties of the specified group.

Requires authentication (administrator ID and password) to use this command.

Syntax

group show *group_name*

group show-dn *dn*

group show-members *group_name*

Options

show *group_name*

Shows the properties of the group that is specified by *group_name*. The group must exist, or an error is displayed.

Examples of group names are `Credit`, `Sales`, and `Test-group`.

show-dn *dn*

Shows the group that is specified by the group identifier in the user registry. The returned group is defined in the user registry, but it is not necessarily a Security Verify Access group. Groups that are not Security Verify Access groups can be imported into Security Verify Access by use of the **group import** command. For example, the format for a distinguished name is like `"cn=engineering,ou=Austin,o=Tivoli,c=us"`.

show-members *group_name*

Lists the user names of the members of the specified group. The group must exist, or an error is displayed.

Examples of group names are `Credit`, `Sales`, and `Test-group`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays properties of the `credit` group:

  ```
  pdadmin sec_master> group show credit
  ```

  The output is like:

  ```
  Group ID: credit
  LDAP dn: cn=credit,ou=Austin,o=Tivoli,c=US
  Description: Credit, Dept HCUS
  ```

```
LDAP cn: credit
Is SecGroup: yes
```

- The following example displays properties that are specified by the identifier of the group, `cn=credit,ou=Austin,o=Tivoli,c=US` in the user registry:

  ```
  pdadmin sec_master> group show-dn cn=credit,ou=Austin,o=Tivoli,c=US
  ```

  The output is like:

  ```
  Group ID: credit
  LDAP dn: cn=credit,ou=Austin,o=Tivoli,c=US
  Description: Credit, Dept HCUS
  LDAP cn: credit
  Is SecGroup: yes
  ```

- The following example lists the user names of the members of the `credit` group:

  ```
  pdadmin sec_master> group show-members credit
  ```

  The output is like:

  ```
  dlucas
  mlucaser
  ```

See also

"group list" on page 49

help

Obtains system help for **pdadmin** commands and options.

This command does not require a login or authentication to use.

Syntax

help {*topic* | *command*}

Options

command
Specifies the miscellaneous command for which help is needed.

topic
Specifies the help command topic for which help is needed.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists help topics and commands:

  ```
  pdadmin> help
  ```

The output is like:

```
Type 'help <topic>' or 'help <command> for more information

Topics:
acl
action
admin
authzrule
config
context
domain
errtext
exit
group
help
login
logout
object
objectspace
policy
pop
quit
rsrc
rsrccred
rsrcgroup
server
user

Miscellaneous Commands:
exit
help
quit
```

- The following example lists the options and descriptions that are available whether you specify the topic `action` or `action create`:

```
pdadmin> help action
```

Or:

```
pdadmin> help action create
```

The output is like:

```
action create <action-name> <action-label> <action-type>
Creates a new ACL action definition
action create <action-name> <action-label> <action-type> <action-group-name>
Creates a new ACL action definition in a group
...
```

login

Establishes authentication credentials that are used for communication with the Security Verify Access policy server. These credentials are used to determine access privileges for the user to policy server data. Most commands cannot be performed unless an explicit login is done.

This command does not require a login or authentication to use.

Syntax

login −a *admin_id* [−p *password*] [−d *domain*]

login −a *admin_id* [−p *password*] [−m]

login −l

Description

Credentials are used to determine user access privileges to policy server data. Except the **context, errtext**, **exit**, **help**, **login**, **logout**, and **quit** commands, and the local configuration commands, a user ID, and a password are needed for authentication.

Credentials are not accumulated or stacked. A **login** command completely replaces any existing credentials.

In interactive mode, the **pdadmin** prompt changes, depending on how the user logs in:

- Not interactive mode. This command starts the **pdadmin** utility. In interactive mode, the **login** commands are entered from the pdadmin> prompt.

  ```
  c:\> pdadmin
  pdadmin>
  ```

- A user local login that is performed for local configuration. No authentication is required.

  ```
  pdadmin> login -l
  pdadmin local>
  ```

- An administrator login that is performed to the local domain. In some cases, the local domain might be the management domain, which is named Default. Authentication is required.

  ```
  pdadmin> login -a sec_master -p secmstrpw
  pdadmin sec_master>
  ```

- A user login that is performed to the local domain. Authentication is required.

  ```
  pdadmin> login -a dlucas -p lucaspw
  pdadmin dlucas>
  ```

- A user login that is performed to another domain other than their local domain. Authentication is required.

  ```
  pdadmin> login -a dlucas -p lucaspw -d domain_a
  pdadmin dlucas@domain_a>
  ```

- A user login that is performed to the management domain. Authentication is required.

  ```
  pdadmin> login -a dlucas -p lucaspw -m
  pdadmin dlucas@Default>
  ```

Options

–a *admin_id*
Specifies an administrator ID.

–d *domain*
Specifies the Security Verify Access secure domain for the login. The *admin_id* user must exist in this domain.

–m
Specifies that the login operation must be directed to the management domain. The *admin_id* user must exist in this domain.

Note: Only one of the following domain options can be specified: –d *domain* or –m. If neither option is specified, the target domain is the local domain that is configured for the system. The *admin_id* user must exist in the target domain, whether it is explicitly specified.

–p *password*
Specifies the password for the *admin_id* user. If this option is not specified, the user is prompted for the password. The password cannot be specified if the *admin_id* is not specified.

-l

Specifies a local login operation. When modifications are made to local configuration files by using the **config** commands, a local login is required before you can run commands. The user can run the **context show** command to view more authentication information.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example logs the `sec_master` user in to the management domain and then displays the authentication context for the user:

  ```
  pdadmin> login -a sec_master -p pa55w0rd -m

  pdadmin sec_master> context show

  User: sec_master
  Domain: Default
  The user is logged in to the management domain.
  ```

- The following example logs in a user to the `domain1` domain and then displays the authentication context for the user:

  ```
  pdadmin> login -a domain1_admin -p d0main1pwd -d domain1

  pdadmin domain1_admin@domain1> context show

  User: domain1_admin
  Domain: domain1
  The user is not logged in to the management domain
  ```

- The following example interactively logs in the user to their local domain that is configured for the system. The domain name is `testdomain`. The example then displays the authentication context of the user:

  ```
  pdadmin> login
  Enter User ID: testdomain_admin
  Enter password: adminpwd

  pdadmin testdomain_admin> context show

  User: testdomain_admin
  Domain: testdomain
  The user is not logged in to the management domain
  ```

- The following example of a local login demonstrates how the prompt changes, depending on the type of interactive login:

  ```
  c:\> pdadmin login -l
  ```

  Provides this prompt:

  ```
  pdadmin local>
  ```

logout

Discards any authentication credentials that are in effect.

This command does not require a login or authentication to use.

Syntax

`logout`

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example first shows a local login and then demonstrates how the prompt changes:

  ```
  pdadmin login -l
  pdadmin local>
  ```

  The following example demonstrates the **`logout`** command:

  ```
  pdadmin local> logout
  ```

- The following example displays:

  - Context information about the user ID.
  - Context information about the domain ID.
  - Whether the domain is a management domain.

  ```
  pdadmin domain1_admin@domain1> context show
  User: domain1_admin
  Domain: domain1
  The user is not logged in to the management domain.
  ```

  The following example shows a **`logout`** command, and then displays context information after the **`logout`** command was issued:

  ```
  pdadmin domain1_admin@domain1> logout
  The user has been logged out and credentials have been discarded.

  pdadmin>context show
  No login information.
  ```

See also

“exit or quit” on page 45
“login” on page 54
“context show” on page 38

object access

Confirms whether the specified access is permitted on the specified object. The access is determined based on the permissions of this user.

Requires authentication (administrator ID and password) to use this command.

Syntax

`object access` *`object_name`* *`permissions`*

Options

object_name

Specifies the protected object, which is the fully qualified name of the object, including the object space within which it is located.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

permissions

Specifies the permission or permissions to check. Security Verify Access uses a set of default actions that cover a wide range of operations. Actions are represented by single alphabetic ASCII characters (a-z, A-Z).

For example, a list of primary action tasks and associated permissions for the user `sec_master`, with WebSEAL as the web server, might include:

```
TTraverseBase
cControlBase
gDelegationBase
mModifyGeneric
dDeleteGeneric
bBrowseBase
sServer AdminGeneric
vViewGeneric
aAttachBase
BBypass POPBase
tTraceBase
rReadWebSEAL
xExecuteWebSEAL
lList DirectoryWebSEAL
NCreateBase
WPasswordBase
AAddBase
RBypass AuthzRuleBase
```

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example confirms whether the user who is running **pdadmin** has the Bypass POP (**B**) permission on the object named `/Management`:

  ```
  pdadmin sec_master> object access /Management B
  ```

  The output is like:

  ```
  Access: No
  ```

- The following example confirms whether the user who is running **pdadmin** has action Password (**W**) permission on the object named `/Management/test-object`:

  ```
  pdadmin sec_master> object access /Management/test-object W
  ```

  The output is like:

  ```
  Access: Yes
  ```

See also

“object listandshow” on page 65
“object show” on page 69

object copy

Recursively copy the policy from one protected object space to another. The policy that is copied includes ACLs, POPs, authorization rules, and extended attributes.

Requires authentication (administrator ID and password) to use this command.

Both source and destination objects must exist under an object space.

The recursive copy occurs in the background and the command will immediately return. Only one copy operation occurs at a time. So extra copy operations are queued until the copy operation that preceded it is completed.

A message is logged in the policy server's log file when the copy operation starts and when it completes. The completion message also indicates whether the copy operation is a success or failure.

Syntax

`object copy` `-recursive` *`src_object_name dst_object_name`*

Options

`src_object_name`

The source object name. Specifies the protected object, which is the fully qualified name of the object, including the object space within which it is located. The *`src_object_name`* and all its child objects are copied to the *`dst_object_name`* including ACLs, POPs, authorization rules, and extended attributes that are attached to them.

An example object name is: `/WebSEAL/abc.ibm.com-default`

`dst_object_name`

The destination object name. Specifies the protected object, which is the fully qualified name of the object, including the object space within which it is located. The *`src_object_name`* and all its child objects are copied to the *`dst_object_name`* including ACLs, POPs, authorization rules, and extended attributes that are attached to them.

An example object name is: `/WebSEAL/new.ibm.com-default`

Return codes

0
: The command completed successfully.

1
: The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

The following example performs a recursive copy of the contents of the `/WebSEAL/abc.ibm.com-default` object to the `/WebSEAL/new.ibm.com-default` object.

```
pdadmin sec_master> object copy -recursive /WebSEAL/abc.ibm.com-default
/WebSEAL/new.ibm.com-default
```

object create

Creates a protected object.

Authentication (administrator ID and password) required to use this command.

Syntax

object create *object_name* *object_description* *type* ispolicyattachable {yes|no}

Options

object_description
: Specifies any text string that describes the object that is being created.

  A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

  An example of a description is "`Travel Groups`".

ispolicyattachable {yes|no}
: Specifies whether an ACL, a protected object policy, or an authorization rule can be attached to this object. Valid values are `yes` or no.

object_name
: Specifies the name for the protected object that is being created. This name is the fully qualified name of the object, including the object space within which it is located. This name must be unique.

  A valid object name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set.

  Examples of object names are:

  - `/Management/Groups/Travel`
  - `/WebSEAL`
  - `/Management`

type
: Specifies the type of object to be created. Types range from 0 to 17. For example, types `10` or 16 are appropriate for container objects. Object types are described in the Administering topics in the IBM Knowledge Center.

You can assign any of the following types:

0
Unknown

1
Secure domain

2
File

3
Executable program

4
Directory

5
Junction

6
WebSEAL server

7
Unused

8
Unused

9
HTTP server

10
Nonexistent object

11
Container object

12
Leaf object

13
Port

14
Application container object

15
Application leaf object

16
Management object

17
Unused

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates the object named `/Management/test-object` that has a description of `Test Object` and is an application container object (14). An ACL or a protected object policy can be attached to this object:

  ```
  pdadmin sec_master> object create /Management/test-object "Test Object" 14
  ispolicyattachable yes
  ```

- The following example creates the object named `/Management/Groups/Travel` that has a description of `Travel Container Object` and is an application container object (14). An ACL or a protected object policy cannot be attached to this object:

  ```
  pdadmin sec_master> object create /Management/Groups/Travel "Travel
  Container Object" 14 ispolicyattachable no
  ```

See also

“object exists” on page 63
“object delete” on page 62

object delete

Deletes a protected object.

Requires authentication (administrator ID and password) to use this command.

Syntax

`object delete` `{-recursive}` *`object_name`*

Options

`-recursive`
Specifies that a recursive delete is performed. This means that all objects under the specified object name are deleted.

`object_name`
Specifies the protected object to be deleted, which is the fully qualified name of the object, including the object space in which it is located. The object must exist, or an error is displayed.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

`-recursive`
Specifies that a recursive delete is performed. This means that all objects under the specified object name are deleted.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the object named `/Management/test-object`:

```
pdadmin sec_master> object delete /Management/test-object
```

- The following example deletes the object named `/Management/Groups/Travel`:

```
pdadmin sec_master> object delete /Management/Groups/Travel
```

See also

"object exists" on page 63
"object create" on page 60

object exists

Indicates whether a protected object exists.

The protected object might be located either in:

- The policy database, or in
- An object space that is managed by an administration service plug-in.

The administration service plug-in might be registered by an authorization application, such as WebSEAL. Use this command to determine whether the specified object was created.

Authentication (administrator ID and password) is required to use this command.

Syntax

`object exists` *`object_name`*

Options

`object_name`

Specifies the protected object, which is the fully qualified name of the object, including the object space within which it is located.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example confirms whether the object named `/Management/protected_object1` exists:

  ```
  pdadmin sec_master> object exists /Management/protected_object1
  ```

  The output is like:

  ```
  Exists: Yes
  ```

- The following example confirms whether the object named `/Management/notAnObject` exists:

  ```
  pdadmin sec_master> object exists /Management/notAnObject
  ```

  The output is like:

  ```
  Exists: No
  ```

See also

"object listandshow" on page 65
"object show" on page 69

object list

Lists any objects that are grouped under the specified protected object. Alternatively, lists all the extended attributes that are associated with the specified protected object.

Requires authentication (administrator ID and password) to use this command.

Syntax

object list

object list *object_name*

object list *object_name* attribute

Options

object list
Lists all protected objects. The output is the same as if you issued the **objectspace list** command.

object list ***object_name***
Lists all objects that are grouped under the specified protected object. The specified object is the fully qualified name of the object, including the object space within which it is located.

object list ***object_name*** **attribute**
Lists all extended attributes that are associated with the specified protected object. The object must exist, or an error is displayed.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists all the protected object spaces under the root of the object namespace (/):

  ```
  pdadmin sec_master> object list
  ```

  Displays a list like:

  ```
  /Management
  /MyObjectSpace_1
  ...
  /WebSEAL
  ```

- The following example lists all the protected objects under the protected object named `/Management`. In this example, both `/Management` and `/Management/ACL` are object spaces:

  ```
  pdadmin sec_master> object list /Management
  ```

  Displays a list like:

  ```
  /Management/ACL
  /Management/Action
  /Management/Config
  ...
  /Management/test-object
  ```

- The following example lists the extended attributes for the object named `/Management/test-object`:

  ```
  pdadmin sec_master> object list /Management/test-object attribute
  ```

  Displays a list of attributes like:

  ```
  test1
  ```

See also

"object listandshow" on page 65
"object show" on page 69

object listandshow

Lists any child objects that are grouped under the specified protected object and displays all values that are associated with each object. Shows all values that are associated with the protected object, including the attached ACLs, POPs, and authorization rules. Also shows any policies that are inherited from protected objects higher in the hierarchy.

Requires authentication (administrator ID and password) to use this command.

Syntax

`object listandshow` *`object_name`*

Options

`object_name`

Specifies the protected object for which the child objects and associated values are to be displayed. The specified protected object is the fully qualified name of the object, including the object space within which it is located.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`

- /Management

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists the object named /Management/Groups/Travel and also automatically lists extended attributes, if any:

  ```
  pdadmin sec_master> object listandshow /Management/Groups/Travel
  ```

  Displays information like:

  ```
  Name : /Management/Groups/Travel
  Description : Travel Container Object
  Type : 14 (Application Container Object)
  Is Policy Attachable : no
  Extended Attributes :
  test1
  1111
  ```

- The following example displays the object named /Management/test-object and lists any attached policies (myrule) and effective policies (myacl and mypop):

  ```
  pdadmin sec_master> object listandshow /Management/test-object
  ```

  Displays information like:

  ```
  Name : /Management/test-object
  Description : Test Object
  Type : 14 (Application Container Object)
  Is Policy Attachable : yes
  Attached ACL :
  Attached POP :
  Attached AuthzRule : myrule
  Effective ACL : myacl
  Effective POP : mypop
  Effective AuthzRule : myrule
  ```

See also

"object list" on page 64
"object show" on page 69

object modify

Modifies an existing object.

Requires authentication (administrator ID and password) to use this command.

Important: User cannot modify objects within a shared object space or underneath an objectspace path, as opposed to under a standard object path.

Syntax

object modify *object_name* delete attribute *attribute_name* [*attribute_value*]

object modify *object_name* set attribute *attribute_name attribute_value*

object modify *object_name* set description *description*

object modify *object_name* set ispolicyattachable {yes|no}

object modify *object_name* set type *type*

Options

delete attribute *attribute_name* [*attribute_value*]

Deletes the specified extended attribute (name and value) from the specified protected object. The attribute must exist, or an error is displayed. When you delete the last value for an attribute, it also deletes the attribute from the specified protected object. The optional *attribute_value* deletes the specified value from the specified extended attribute key in the specified protected object. Examples of attribute names and values:

```
test11111
Dept_No445
Employee_name"Diana Lucas"
```

object_name

Specifies the protected object to be modified. The specified protected object is the fully qualified name of the object, including the object space within which it is located. The object must exist, or an error is displayed.

Examples of object names are:

- /Management/Groups/Travel
- /WebSEAL
- /Management

set attribute *attribute_name attribute_value*

Creates an extended attribute, with the specified name and value, and adds it to the specified protected object. If the attribute exists, the attribute value is added as an additional value if the same value does not exist for this attribute. If the same value exists for this attribute, it does not get added again (duplicate values are not allowed), and no error is returned.

The optional *attribute_value* sets the specified value from the specified extended attribute key in the specified protected object. The attribute value must exist, or an error is displayed.

Examples of extended attribute names and values:

```
attr1valueA
attr1valueB
attr2valueC
```

set description *description*

Sets the description field of the specified protected object.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are allowed. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: "Travel Group aaa"

set ispolicyattachable {yes|no}

Sets whether the protected object can have an ACL, a POP, or an authorization rule attached or not. Valid values are yes or no.

set type *type*

Specifies the type of the object space to be created. Types range from 0 to 17. For example, types 10 or 16 are appropriate for objects.

You can assign any of the following types:

0
Unknown

1
Secure domain

2
File

3
Executable program

4
Directory

5
Junction

6
WebSEAL server

7
Unused

8
Unused

9
HTTP server

10
Nonexistent object

11
Container object

12
Leaf object

13
Port

14
Application container object

15
Application leaf object

16
Management object

17
Unused

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example sets the `ispolicyattachable` option for the `/Management/Groups/Travel` object:

  ```
  pdadmin sec_master> object modify /Management/Groups/Travel set
  ispolicyattachable yes
  ```

- The following example sets the attributes for the `/Management/test-object` object:

  ```
  pdadmin sec_master> object modify /Management/test-object set attribute
  test1 1111
  ```

See also

"object create" on page 60

object show

Shows values for the protected object.

If the protected object name specified does not exist, default values are shown. To determine whether a protected object exists, use the **`object show`** command.

Requires authentication (administrator ID and password) to use this command.

Syntax

`object show` *`object_name`* [`attribute` *`attribute_name`*]

Description

The **`object show`** command shows values that are associated with the protected object.

The object values shown can include:

- ACLs.
- POPs.
- Authorization rules.
- Extended attributes, such as attribute name and value pairs.

These extended attributes can be attached directly to the object or inherited from protected objects in the hierarchy of this object.

When the **attribute** option is specified, the **attribute_name** value or values are shown if the attribute is attached to the protected object specified.

This command limits the output for POPs, ACLs, and authorization rules, which are based on the permissions of the user. A user must have the view (**v**) permission on the object to show it.

Options

`object_name`

Specifies the protected object. The specified protected object is the fully qualified name of the object, including the object space within which it is located.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

attribute *attribute_name*

Specifies the name of the extended attribute whose values are to be displayed. (Optional) The extended attribute must exist for the object name that is specified, or an error is displayed. In the example that is listed for the /object-text object in “Examples” on page 70, the following extended attributes are shown:

- test1
- test2
- abc

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the /object-test object and lists all attached and effective ACLs, POPs, authzrules, and extended attributes:

  ```
  pdadmin sec_master> object show /object-test
  ```

  Displays information like:

  ```
  Name: /object-test
  Description: Test object
  Type: 12 (Leaf Object)
  Is Policy Attachable : Yes
  Extended Attributes:
  Name:test1
  Value(s): 1111
  Name:test2
  Value(s): abc
  2222
  second
  Attached ACL:
  Attached POP:
  Attached AuthzRule:

  Effective Extended Attributes:
  Protected Object Location: /object-test
  Name:test1
  Value(s): 1111
  Name:test2
  Value(s): abc
  2222
  second
  Effective ACL: default-root
  Effective POP:
  Effective AuthzRule:
  ```

- The following example displays the /object-test/child1 object and lists all attached and effective ACLs, POPs, AuthzRules, and extended attributes:

  ```
  pdadmin sec_master> object show /object-test/child1
  ```

  Displays information like:

  ```
  Name: /object-test/child1
  Description: Child 1
  Type: 12 (Leaf Object)
  Is Policy Attachable : Yes
  Extended Attributes:
  ```

```
Attached ACL:
Attached POP:
Attached AuthzRule:

Effective Extended Attributes:
Protected Object Location: /object-test
Name:test1
Value(s): 1111
Name:test2
Value(s): abc
2222
second
Effective ACL: default-root
Effective POP:
Effective AuthzRule:
```

- The following example displays information about the `test1` attribute that is listed for `object/object-test/child1`:

  ```
  pdadmin sec_master> object show /object-test/child1 attribute test1
  ```

  Because the `test1` attribute is an extended attribute of the `/object-test` object, the command returns the following message:

  ```
  Could not perform the administration request
  Error: HPDAC0463E There are no extended attributes associated with
  the specified protected object or authorization policy object. (status
  0x1005b1cf)
  ```

  To view the information about the `test1` attribute of the `/object-test` object, enter the following command:

  ```
  pdadmin sec_master> object show /object-test attribute test1
  ```

  Displays information like:

  ```
  test1
  1111
  ```

- The following example displays the `/Management/test-object` object, which lists any attached (`myrule`) and effective (`myacl` and `mypop`) policies:

  ```
  pdadmin sec_master> object show /Management/test-object
  ```

  Displays information like:

  ```
  Name: /Management/test-object/
  Description : Test object
  Type: 14 (Application Container Object)
  Is Policy Attachable: Yes
  Extended Attributes:
  Attached ACL: myacl
  Attached POP: mypop
  Attached AuthzRule: myrule

  Effective Extended Attributes:
  Effective ACL: myacl
  Effective POP: mypop
  Effective AuthzRule: myrule
  ```

- The following example creates a protected object and then performs an **`object show`** of that protected object. An **`object show`** is then performed for an object that has not been created. Then the **`object exists`** command is issued for both of these objects.

  ```
  pdadmin sec_master> object create /Management/new_object1" "0ispoly

  pdadmin sec_master> object show /Management/new_object1
  Name: /Management/new_object1
  Description:
  Type: 0 (Unknown)
  Is Policy Attachable: Yes
  Extended Attributes:
  ```

```
Attached ACL:
Attached POP:
Attached AuthzRule:

Effective Extended Attributes:
Effective ACL: default-management
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /Management/not_there_object
Name: /Management/not_there_object
Description:
Type: 0 (Unknown)
Is Policy Attachable: Yes
Extended Attributes:
Attached ACL:
Attached POP:
Attached AuthzRule:

Effective Extended Attributes:
Effective ACL: default-management
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object exists /Management/new_object1
Exists: Yes
pdadmin sec_master> object exists /Management/not_there_object
Exists: No
```

See also

"object list" on page 64
"object list" on page 64
"object listandshow" on page 65

objectspace create

Creates a protected object space under which protected objects can be placed.

Requires authentication (administrator ID and password) to use this command.

Syntax

`objectspace create` *`objectspace_name description type`*

Description

The root of the new protected object space automatically has the `ispolicyattachable` option that is set to `true`.

Options

`description`

Specifies the description of the new object space.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

An example description is `"Accounting"`.

`objectspace_name`

Specifies the name of the object space to be created.

A valid object space name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set.

Examples of object space names are `/Management` and `/WebSEAL`.

type

Specifies the type of the object space to be created. Types range from 0 to 17. For example, types 10 or 16 are appropriate for objects and object spaces.

You can assign any of the following types:

0
Unknown

1
Secure domain

2
File

3
Executable program

4
Directory

5
Junction

6
WebSEAL server

7
Unused

8
Unused

9
HTTP server

10
Nonexistent object

11
Container object

12
Leaf object

13
Port

14
Application container object

15
Application leaf object

16
Management object

17
Unused

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates an object space named /Test-Space that is an application container object (type 14):

```
pdadmin sec_master> objectspace create /Test-Space "New Object Space" 14
```

- The following example creates an object space named /Dept4D4 that is a management object (type 16):

```
pdadmin sec_master> objectspace create /Dept4D4 "Department 4D4" 16
```

See also

"objectspace delete" on page 74

objectspace delete

Deletes the specified protected object space.

Requires authentication (administrator ID and password) to use this command.

Syntax

objectspace delete *objectspace_name*

Options

objectspace_name

Specifies the name of the object space to be deleted. The object space must exist or an error is displayed.

Examples of object space names are /Management and /WebSEAL.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the object space named /Test-Space:

```
pdadmin sec_master> objectspace delete /Test-Space
```

- The following example deletes the object space named /Dept4D4:

```
pdadmin sec_master> objectspace delete /Dept4D4
```

See also

"objectspace create" on page 72

objectspace list

Lists all the existing protected object spaces in the policy server.

Requires authentication (administrator ID and password) to use this command.

Syntax

objectspace list

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists all the protected object spaces:

```
pdadmin sec_master> objectspace list
```

Displays a list like:

```
/Management
/MyObjectSpace_1
...
/WebSEAL
```

policy get

Displays the policy for user passwords, account rules, and conditions. Requires authentication (administrator ID and password) to use this command.

Syntax

policy get account-expiry-date [–user *user_name*]

policy get disable-time-interval [–user *user_name*]

policy get max-concurrent-web-sessions [–user *user_name*]

policy get max-login-failures [–user *user_name*]

policy get max-password-age [–user *user_name*]

policy get max-password-repeated-chars [–user *user_name*]

policy get min-password-alphas [–user *user_name*]

policy get min-password-length [–user *user_name*]

policy get min-password-non-alphas [–user *user_name*]

policy get password-spaces [–user *user_name*]

policy get `tod-access` [–user *user_name*]

Options

–user *user_name*
Specifies the user whose policy information is to be displayed. If this option is not specified, the general policy is displayed. For any specified policy, if a user has a specific policy that is applied, this specific policy takes precedence over any general policy that might also be defined. The precedence applies regardless of whether the specific policy is more or less restrictive than the general policy. Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`. (Optional)

account-expiry-date
Displays the account expiration date.

disable-time-interval
Displays the time, in seconds, to disable user accounts when the maximum number of login failures is exceeded.

max-concurrent-web-sessions
Displays the maximum number of concurrent web sessions. The value is a number equal to or greater than 1 or one of the following values:

displace
All existing web sessions end when the user starts a new web session.

unlimited
The user can start an unlimited number of web sessions.

unset
The web session policy is not set.

This policy applies only to certain components. A *web session* is a user session that is maintained by a web security solution, such as WebSEAL or the plug-in for web servers. See the IBM Knowledge Center to determine whether this setting is applicable and whether specific configuration options are required to enforce this policy.

max-login-failures
Displays the maximum number of login failures. To enforce maximum login failures, the **disable-time-interval** parameter must be set. For more information, see the **disable time interval** section.

max-password-age
Displays the maximum time that a password is valid. The time is indicated in days, expressed as `000–00:00:00`. For example, `31-08:30:00` for 31 days, 8 hours, 30 minutes, 0 seconds. This time is relative to the last time the password was changed.

max-password-repeated-chars
Displays the maximum number of repeated characters that are allowed in a password.

min-password-alphas
Displays the minimum number of alphabetic characters that are required in a password.

min-password-length
Displays the minimum password length.

min-password-non-alphas
Displays the minimum number of non-alphabetic characters that are required in a password.

password-spaces
Displays whether spaces are allowed in passwords.

tod-access
Displays the time of day access policy.

Return codes

0
The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example returns the account expiration date of unlimited for the specified user dlucas:

  ```
  pdadmin sec_master> policy get account-expiry-date -user dlucas
  Account expiry date: unlimited
  ```

- The following example returns the maximum time of 0 days, where zero indicates unlimited, that the password is valid for the specified user dlucas:

  ```
  pdadmin sec_master> policy get max-password-age -user dlucas
  ```

  For unlimited password age, returns information like:

  ```
  Maximum password age: 0-0:0:0
  ```

See also

"policy set" on page 77

policy set

Sets the policy for user passwords, account rules, and conditions. Requires authentication (administrator ID and password) to use this command.

Syntax

policy set account-expiry-date {unlimited|*absolute_time*|unset} [–user *user_name*]

policy set disable-time-interval {*number*|unset|disable} [–user *user_name*]

policy set max-concurrent-web-sessions {*number*|displace|unlimited|unset} [–user *user_name*]

policy set max-login-failures {*number*|unset} [–user *user_name*]

policy set max-password-age {unset|*relative_time*} [–user *user_name*]

policy set max-password-repeated-chars {*number*|unset} [–user *user_name*]

policy set min-password-alphas {unset|*number*} [–user *user_name*]

policy set min-password-length {unset|*number*} [–user *user_name*]

policy set min-password-non-alphas {unset|*number*} [–user *user_name*]

policy set password-spaces {v|no|unset} [–user *user_name*]

policy set tod-access {{anyday|weekday|*day_list*}:{anytime|*time_spec*} [:{utc|local}]|unset} [–user *user_name*]

Description

The valid range for numbers can be any number. However, use a reasonable number for the task that you want to complete. For example, a minimum password length must be long enough to protect your system. In addition, the password must not be so short as to make it easy for someone to determine your password by trying different combinations.

When you define the password policy, ensure that this definition complies with the password policy of the underlying operating systems and user registries.

Options

`account-expiry-date` {`unlimited`|*`absolute_time`*|`unset`}

Sets the account expiration date. The *`absolute_time`* format is specified in the following format:

```
YYYY-MM-DD-hh:mm:ss
```

The hours must be entered by using a 24-hour clock (for example, 09 for 9 a.m. or 14 for 2 p.m.). The default value is `unset`.

If you set the account expiration date, it is set for all accounts that do not use the `–user` *`user_name`* option. By default, the **sec_master** user account has a per-user account expiration date of unlimited. If you set the account expiration date to unlimited, do the following actions:

- Set `max-password-age` to `0` for unlimited.
- Set `tod-access` to `anyday:anytime:local`.
- Use the `–user` *`user_name`* option.

`disable-time-interval` {*number*|`unset`|`disable`}

Sets the time, in seconds, to disable each user account when the maximum number of login failures is exceeded. Security Verify Access does not impose an upper limit for the maximum number allowed. Use a range from 0 (unlimited) to a number that represents the value that is most logical for the parameter you are trying to set. The default value is 180 seconds.

`max-concurrent-web-sessions` {*number*|`displace`|`unlimited`|`unset`}

Sets the maximum number of concurrent web sessions. This policy applies only to certain components. A *web session* is a user session that is maintained by a web security solution, such as WebSEAL or the plug-in for web Servers. See the IBM Knowledge Center to determine whether this setting is applicable and whether specific configuration options are required to enforce this policy.

This option supports the following values:

number

Specifies the maximum number of concurrent web sessions that can be established. This value is a number that is equal to or greater than one.

`displace`

Specifies that if a user starts a new web session, any existing web session ends.

`unlimited`

Allows unlimited concurrent web sessions.

`unset`

Specifies to unset concurrent web session policy.

`max-login-failures` {*number*|`unset`}

Sets the maximum number of login failures allowed. Security Verify Access does not impose an upper limit for the maximum number allowed. Instead, use a range from zero to a number that represents the value that is most logical for the parameter you are trying to set. If the number is too large, it might render the login policy ineffective. The default value is 10.

To enforce maximum login failures, the `disable-time-interval` parameter must be set. See **disable-time-interval** for more information about `disable-time-interval`.

`max-password-age` {`unset`|*`relative_time`*}

Sets the maximum time, in days, that a password is valid. This policy is a global password policy as opposed to the individual user policy. The individual user policy:

- Is set by using the **`user modify`** command with the *`user_name password-valid`* option.
- Enables or disables the validity of a password for the specified user account.

The *relative_time* option is relative to the number of days since the last password change occurred. The *relative_time* format is specified in the following format:

```
DDD-hh:mm:ss
```

The valid range is from `000–00:00:00` to `999–23:59:59`. A value of zero (`000–00:00:00`) indicates that the password never expires. The default value is `91 days`. This value is expressed as `91–00:00:00`.

max-password-repeated-chars {*number*|unset}

Sets the maximum number of consecutively, repeated characters that are allowed in a password. Security Verify Access does not impose an upper limit on the maximum number allowed. Instead, use a range from 0 to a number that represents the most logical value for the parameter you are trying to set. If the number is too large, it might render the password policy ineffective. The default value is 2.

Example: If `max-password-repeated-chars` is set to 2, then `password` and `pspassword` are both valid values. However, `passsword` is not valid because the character `s` occurs three times consecutively.

min-password-alphas {unset|*number*}

Sets the minimum number of alphabetic characters that are required in a password. Security Verify Access does not impose an upper limit for the minimum number allowed. Instead, use a number that represents the value that is most logical for the parameter you are trying to set. If the number is too small, it might render the password policy ineffective. The default value is 4.

min-password-length {unset|*number*}

Sets the minimum password length. Security Verify Access does not impose an upper limit for the minimum number allowed. Instead, use a number that represents the value that is most logical for the parameter you are trying to set. If the number is too large, the password policy might be difficult to adhere to. The default value is 8.

min-password-non-alphas {unset|*number*}

Sets the minimum number of non-alphabetic characters that are required in a password. Security Verify Access does not impose an upper limit for the minimum number allowed. Instead, use a number that represents the value that is most logical for the parameter you are trying to set. If the number is too large, the password policy might be difficult to adhere to. The default value is 1.

password-spaces {v|no|unset}

Sets the policy of whether spaces are allowed in passwords. The default value is `unset`.

tod-access {{anyday|weekday|*day_list*}:{anytime|*time_spec*} [:{utc|local}]|unset}

Sets the time of day access policy.

The *day_list* is a comma-separated list of days of the week, each of which is represented by a three-character value (for example, `mon,wed,fri`). The *day_list* specifies which days of the week you can log in to the account. If you want to list every day of the week, specify `anyday`; if you do not want to include the weekend days, specify `weekday`.

The *time_spec* format is specified in the following format:

```
hhmm
```

The format is expressed by using a 24-hour clock. For example, `0900` for 9 a.m. or `1430` for 2:30 p.m. The default value is `unset`, and the optional time zone is `local` by default. The *time_spec* value and time zone specify the time of day when you can log in to the account.

Note:

- `utc=GMT`
- When you modify a password policy, you provide a list of days, start time, and end time. The start time and end time apply to each day on the list. If the specified start time is greater than the specified end time, then the access is allowed until the specified end time of the next day.

-user ***user_name***

Specifies the user whose policy information is to be set. If this option is not specified, the general policy is set. For any specified policy, if a user has a specific policy that is applied, this specific policy takes precedence over any general policy that might also be defined. The precedence applies regardless of whether the specific policy is more or less restrictive than the general policy.

A valid user name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`. (Optional)

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example sets the account expiration date of December 30, 1999, at 11:30 p.m. for the specified user `dlucas`:

```
pdadmin sec_master> policy set account-expiry-date 1999-12-30-23:30:00
-user dlucas
```

- The following example sets the maximum password age of 31 days, 8 hours, 30 minutes, and 0 seconds for the specified user `dlucas`:

```
pdadmin sec_master> policy set max-password-age 031-08:30:00 -user dlucas
```

- The following example sets the maximum of 12 concurrent web sessions:

```
pdadmin sec_master> policy set max-c 12
```

See also

"policy get" on page 75

pop attach

Attaches a protected object policy (POP) to the specified protected object. The POP must be created before it can be attached.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop attach *object_name pop_name*

Description

At most, one POP can be attached to a given protected object. If the object already has a POP attached to it, the specified POP replaces the existing one. The same POP can be attached to multiple protected objects. Ensure that the protected object exists in the protected object space before you attempt to attach a POP.

Options

object_name
Specifies the name of the protected object to which the protected object policy is attached. The object must exist, or an error is displayed.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

pop_name
Specifies the name of the protected object policy to be attached. The POP must exist, or an error is displayed.

Examples of POP names are `poptest` and `pop1`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example attaches the POP `pop1` to the protected object named `/Management/test-object`:

```
pdadmin sec_master> pop attach /Management/test-object pop1
```

- The following example attaches the POP `poptest` to the protected object named `/Test-Space`:

```
pdadmin sec_master> pop attach /Test-Space poptest
```

See also

"pop create" on page 81
"pop detach" on page 83

pop create

Creates a protected object policy (POP).

Requires authentication (administrator ID and password) to use this command.

Syntax

pop create *pop_name*

Options

pop_name
Specifies the name of the POP to be created. A valid protected object policy name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed. Avoid the following characters in the POP name:

```
! " # & ( ) * + , ; : < > = @ / \ | .
```

Note: Although a POP name can contain 1 or more of these characters, the results of using such a POP are undefined.

Examples of POP names are `poptest` and `pop1`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example shows how to create and display a POP:

```
pdadmin sec_master> pop create test
```

The new POP contains new POP settings like:

```
pdadmin sec_master> pop show test

Protected object policy:test
Description:
Warning:no
Audit Level:none
Quality of protection: none
Time of day access: sun, mon, tue, wed, thu, fri, sat:
anytime: local
IP Endpoint Authentication Method Policy
Any Other Network 0
```

See also

“pop attach” on page 80
“pop delete” on page 82

pop delete

Deletes the specified protected object policy (POP).

Requires authentication (administrator ID and password) to use this command.

Syntax

pop delete *pop_name*

Options

pop_name
Specifies the name of the POP to be deleted. The POP must exist, or an error is displayed. Examples of POP names are `poptest` and `pop1`.

Return codes

0
The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the POP pop1:

```
pdadmin sec_master> pop delete pop1
```

- The following example deletes the POP `poptest`:

```
pdadmin sec_master> pop delete poptest
```

See also

"pop create" on page 81
"pop detach" on page 83

pop detach

Detaches a protected object policy (POP) from the specified protected object.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop detach *object_name*

Options

object_name

Specifies the protected object from which the POP is to be detached. The object must exist and have a POP attached, or an error is displayed.

Examples of object names are:

- `/Management/Groups/Travel`
- `/WebSEAL`
- `/Management`

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example detaches all POPs from the protected object named `/Management/test-object`:

```
pdadmin sec_master> pop detach /Management/test-object
```

- The following example detaches all POPs from the protected object named `/Test-Space`:

```
pdadmin sec_master> pop detach /Test-Space
```

See also

"pop attach" on page 80
"pop delete" on page 82

pop find

Finds and lists all protected objects that have a protected object policy (POP) attached.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop find *pop_name*

Description

A user must have the browse (**b**) and view (**v**) permissions for the object to be listed when the **object show** command is issued. Otherwise, an error is returned:

```
The user is not authorized to view one or more protected objects where the
requested POP is attached.
```

Options

pop_name
Specifies the name of the POP for which to search. The POP must exist, or an error is displayed. Examples of POP names are `poptest` and `pop1`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example finds all objects to which the POP `pop1` is attached:

```
pdadmin sec_master> pop find pop1

/Management/test-object
```

- The following example finds all objects to which the POP `poptest` is attached:

```
pdadmin sec_master> pop find poptest

/Test-Space
```

See also

"pop list" on page 85

pop list

Lists all protected object policies that are created. Alternatively, lists all extended attributes that are associated with a protected object policy.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop list [*pop_name* attribute]

Options

***pop_name* attribute**

Specifies the POP for which to list the attributes. The POP must exist, or an error is displayed. (Optional)

Examples of POP names are `poptest` and `pop1`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example shows how to list all POPs:

```
pdadmin sec_master> pop list

test
pop1
poptest
```

- The following example shows how to list all the attributes for the POP named `pop1`:

```
pdadmin sec_master> pop list pop1 attribute

attr1
```

See also

"pop find" on page 84

pop modify

Modifies protected object policies.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop modify *pop_name* delete attribute *attribute_name* [*attribute_value*]

pop modify *pop_name* set attribute *attribute_name* *attribute_value*

pop modify *pop_name* set audit-level {all|none|*audit_level_list*}

pop modify *pop_name* set description *description*

pop modify *pop_name* set ipauth add *network netmask level*

pop modify *pop_name* set ipauth anyothernw *level*

pop modify *pop_name* set ipauth remove *network* netmask

pop modify *pop_name* set qop {none|integrity|privacy}

pop modify *pop_name* set tod-access {anyday|weekday|*day_list*}:{anytime|*time_spec-time_spec*}[:{utc|local}]

pop modify *pop_name* set warning {yes|no}

Description

The **pop modify** command modifies a protected object policy (POP). When you use the set ipauth add or set ipauth remove options, you can specify the IP addresses. The values for the *network* and netmask options are TCP/IP addresses. These IP addresses can be specified by using either version 4 (IPv4) or version 6 (IPv6) notation. Both the *network* and netmask options must be specified in the same IP version.

Note: When you use IPv6 notation, do not use prefix notation when you specify IP addresses.

When you specify IP addresses, be aware of the following restrictions:

- For administration commands, IPv4 clients must provide addresses in IPv4 format even with IPv6 servers.
- For C APIs, IPv4 clients must provide addresses in IPv4 format even with IPv6 servers.
- For C APIs, IPv6 clients can provide addresses in IPv4 or IPv6 format to IPv6 servers.
- For Java™ methods, IPv4 and IPv6 clients must provide addresses in IPv4 format to IPv4 servers.
- For Java methods, IPv4 clients can provide addresses in IPv4 or IPv6 format to IPv6 servers.

For an IPv6 address to be accepted (commands, C APIs, and Java methods), the server must be IPv6. You cannot provide an IPv6 address to an IPv4 server.

The operating system functions that are provided to Security Verify Access have certain limitations. Regardless of C or Java clients, IPv4 addresses must be in IPv4 format when you add addresses to a POP.

Options

delete attribute *attribute_name* [*attribute_value*]

Deletes the specified value from the specified extended attribute key in the specified POP. The attribute must exist, or an error is displayed.

The optional *attribute_value* deletes the specified value from the specified extended attribute key in the specified POP.

Examples of extended attribute names and values:

```
Dept_No445
Employee_Name"Diana Lucas"
```

pop_name

Specifies the name of the protected object policy to be modified. The POP must exist, or an error is displayed.

set attribute *attribute_name attribute_value*

Sets or modifies the specified value from the specified extended attribute key in the specified POP. If the attribute exists, the attribute value is added as an additional value if the same value does not exist for this attribute. If the same value exists for this attribute, it does not get added again (duplicate values are not allowed), and no error is returned.

The *attribute_value* sets the specified value from the specified extended attribute key in the specified POP.

Example: `"Credit Card"`

set audit-level {all|none|*audit_level_list*}

Sets the audit level for the specified POP. The format of an *audit_level_list* is a comma-separated list that contains one or more of these audit levels: `permit,deny,error,admin`.

set description *description*

Sets the description of the specified POP.

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. You can specify an empty string ("") to clear an existing description.

Example of description: `"Policies of Jenson Corp."`

set ipauth add *network netmask level*

Sets the IP endpoint authentication settings in the specified POP. The values for the *network* and *netmask* options are TCP/IP addresses. These IP addresses can be specified by using either version 4 (IPv4) or version 6 (IPv6) addresses. Both the *network* and *netmask* options must be specified in the same IP version.

The following values are supported for *level*:

forbidden

A value that prohibits object access.

integer_values

Application-specific integer values that define the step-up authentication levels. All integer values, except `1000`, are supported. For more information about step-up authentication, see the Administering topics in the IBM Knowledge Center.

set ipauth anyothernw *level*

Sets the `anyothernw` (any other network setting) for the IP authentication level in the specified POP. If controlling access by IP address is not important, use the `anyothernw` option to set the authentication level for:

- All IP addresses, and
- IP address ranges not listed explicitly in the POP.

The following values are supported for *level*:

forbidden

A value that prohibits object access.

integer_values

Application-specific integer values that define the step-up authentication levels. All integer values, except `1000`, are supported. For more information about step-up authentication, see the Administering topics in the IBM Knowledge Center.

set ipauth remove *network* netmask

Removes the IP endpoint authentication settings from the specified POP. The values for the *network* and `netmask` options are TCP/IP addresses. These IP addresses can be specified by using either version 4 (IPv4) or version 6 (IPv6) notation. Both the *network* and `netmask` options must be specified in the same IP version.

set qop {none|integrity|privacy}

Sets the quality of protection level for the specified POP. The following string values are supported:

- `none`
- `integrity`
- `privacy`

set tod-access {anyday|weekday|*day_list*}:{anytime|*time_spec-time_spec*}[:{utc|local}]
Sets the time of day range for the specified protected object policy.

The *day_list* is a comma-separated list of days of the week, each of which is represented by a three-character value (for example, mon,wed,fri). The *day_list* specifies which days of the week the object can be accessed. If you want to list every day of the week, specify anyday; if you do not want to include the weekend days, specify weekday.

The *time_spec* format is specified as *hhmm* and is expressed by using a 24-hour clock (for example, 0900 for 9 a.m. or 1430 for 2:30 p.m.). The default value is not defined, and the optional time zone is local by default. The *time_spec* value and time zone specify the time of day the object can be accessed.

Note: utc=GMT

set warning {yes|no}
Sets the warning mode for the specified protected object policy. Valid values are yes or no.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- This example shows how to modify the description for the POP named test:

```
pdadmin sec_master> pop modify test description "Test POP"
```

- This example shows how to turn on the warning mode for the POP named test:

```
pdadmin sec_master> pop modify test set warning yes
```

- This example shows how to set the audit level to audit all requests on a protected object that result in successful:
  - Access by using permit.
  - Denial of access by using deny.

```
pdadmin sec_master> pop modify test set audit-level permit,deny
```

- This example shows how to set an attribute named attr1 with a value of valueA for the POP named pop1:

```
pdadmin sec_master> pop modify pop1 set attribute attr1 valueA
```

See also

“pop attach” on page 80
“pop create” on page 81

pop show

Shows details about the protected object policy (POP). Alternatively, displays the values for the specified extended attribute from the specified protected object policy.

Requires authentication (administrator ID and password) to use this command.

Syntax

pop show *pop_name* [attribute *attribute_name*]

Options

attribute *attribute_name*
: Specifies the name of the extended attribute whose values you want to display. The attribute must exist, or an error is displayed. Examples of attribute names are `Dept_No` and `Employee_Name`. (Optional)

pop_name
: Specifies the POP to display. The POP must exist, or an error is displayed. Examples of POP names are `poptest` and pop1.

Return codes

0
: The command completed successfully.

1
: The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example shows how to show POP information, including the description:

```
pdadmin sec_master> pop show test

Protected object policy:test
Description:Test POP
Warning:no
Audit level:none
Quaility of protection:none
Time of day access: sun, mon, tue, wed, thu, fri, sat:
anytime: local
IP Endpoint Authentication Method Policy
Any Other Network 0
```

- The following example shows attribute `attr1` information for the POP named pop1:

```
pdadmin sec_master> pop show pop1 attribute attr

attr1
valueA
```

See also

“pop find” on page 84
“pop list” on page 85

rsrc create

Creates and names a web server single sign-on resource.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrc create *resource_name* [–desc *description*]

Description

A *web resource* is a web server that serves as the back-end of a WebSEAL GSO-enabled junction. The web resource name must be specified with the –T option when the GSO-enabled WebSEAL junction is created.

Options

–desc *description*

Specifies a description for the resource. Descriptions containing a space must be enclosed in double quotation marks. (Optional)

A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks.

Examples of descriptions are “Engineering Web server – Room 4807” and “Printer in room 345, Bldg 2”.

resource_name

Specifies the name of the resource to be created.

A valid resource name is an alphanumeric string that is not case-sensitive. If the resource is a GSO resource, certain characters are not allowed. See “Characters disallowed for GSO names” on page 188 for the list of these characters.

Examples of resource names are engwebs01 and JonesData.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example is entered as one line. This example creates and names a web resource engwebs01 with an associated description "Engineering Web server – Room 4807":

```
pdadmin sec_master> rsrc create engwebs01 -desc "Engineering Web
server - Room 4807"
```

- The following example is entered as one line. This example creates and names a printer resource "Mary Jones Printer" with an associated description "Printer in room 345, Bldg 2":

```
pdadmin sec_master> rsrc create "Mary Jones Printer" -desc "Printer in
room 345, Bldg 2"
```

See also

“rsrc delete” on page 91

rsrc delete

Deletes the specified single sign-on resource.

Requires authentication (administrator ID and password) to use this command.

Syntax

`rsrc delete` *`resource_name`*

Options

`resource_name`

Specifies the name of the resource to be deleted. The resource must exist, or an error is displayed.

Examples of resource names are `engwebs01` and `JonesData`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the named resource `engwebs01`:

  ```
  pdadmin sec_master> rsrc delete engwebs01
  ```

- The following example deletes the named resource `"Mary Jones Printer"`:

  ```
  pdadmin sec_master> rsrc delete "Mary Jones Printer"
  ```

See also

“rsrc create” on page 90

rsrc list

Returns a list of all the single sign-on resource names.

Requires authentication (administrator ID and password) to use this command.

Syntax

`rsrc list`

Options

None.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example returns a list of all the single sign-on web resource names:

```
pdadmin sec_master> rsrc list
```

The output is like:

```
engwebs01
Mary Jones Printer
```

See also

“rsrc create” on page 90

rsrc show

Displays the resource information for the named resource.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrc show *resource_name*

Options

resource_name
Specifies the name of the resource for which information is shown. The resource must exist, or an error is displayed.

Examples of resource names are `engwebs01` and `JonesData`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example returns information for the specified resource `engwebs01`:

  ```
  pdadmin sec_master> rsrc show engwebs01
  ```

  The output is like:

```
Web Resource Name: engwebs01
Description: Engineering Web server - Room 4807
```

- The following example returns information for the specified resource `"Mary Jones Printer"`:

```
pdadmin sec_master> rsrc show "Mary Jones Printer"
```

Output is like:

```
Web Resource Name: Mary Jones Printer
Description: Printer in room 345, Bldg 2
```

See also

"rsrc list" on page 91

rsrccred create

Creates a single sign-on credential.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrccred create *resource_name* rsrcuser *resource_userid* rsrcpwd *resource_password* rsrctype {web|group} user *user_name*

Description

A resource credential is a credential that is used to identify the authentication information of a user. WebSEAL uses the authentication information when it accesses a back-end web resource or resource group through a GSO-enabled junction. WebSEAL accesses these resources on behalf of that user.

For example, a user `dlucas` might require the authentication identity `4807ws01` and password `pwd4lucas` when accessing the `engwebs01` web resource that is junctioned through WebSEAL.

A resource credential can be created with this authentication information. Then, WebSEAL automatically uses this information to access the `engwebs01` server whenever the user `dlucas` accesses that resource.

Options

resource_name
Specifies the name that is given to the resource or resource group when the resource or resource group was created. The resource or resource group must exist to create the resource credential. If the resource or resource group does not exist or is not specified, an error message is displayed.

Examples of resource names are `engwebs01` and `"Mary Jones Printer"`.

rsrcpwd ***resource_password***
Specifies the password for a user at the web server.

rsrctype {web|group}
Specifies whether the resource type named is `web` (resource) or `group` (resource group).

rsrcuser ***resource_userid***
Specifies the unique user identification (user ID) for the user at the web server.

Examples of user identifications are `4807ws01` and `userD4D`.

user ***user_name***
Specifies the name of the user for whom the resource credential information applies. If the user does not exist or is not specified, an error message is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates the web resource credential named `engwebs01` for the resource user ID `4807ws01` and password `pwd4lucas` given to user `dlucas`:

```
pdadmin sec_master> rsrccred create engwebs01 rsrcuser 4807ws01
rsrcpwd pwd4lucas rsrctype web user dlucas
```

- The following example creates the group resource credential named `printerusers` for the resource user ID `userD4D` and password `pwd4mjones` given to user `"Mary Jones"`:

```
pdadmin sec_master> rsrccred create printerusers rsrcuser userD4D rsrcpwd
pwd4mjones rsrctype group user "Mary Jones"
```

See also

“rsrccred delete” on page 94
“rsrccred modify” on page 96

rsrccred delete

Deletes a single sign-on credential.

Requires authentication (administrator ID and password) to use this command.

Syntax

`rsrccred delete` *`resource_name`* `rsrctype` {web|group} `user` *`user_name`*

Options

`resource_name`
Specifies the name that is given to the resource or resource group when the resource was created. The resource or resource group must exist, or an error is displayed.

Examples of resource names are `engwebs01` and `"Mary Jones Printer"`.

`rsrctype` {web|group}
Specifies whether the resource type named is `web` (resource) or `group` (resource group) for the single sign-on resource that is associated with the credential. The type of resource must match the resource type that is assigned when the resource or resource group was first created.

`user` ***`user_name`***
Specifies the name of the user for whom the resource credential information applies. The user must exist, or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0
The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example deletes the resource credential information for the resource engwebs01, resource type web, and user name dlucas:

  ```
  pdadmin sec_master> rsrccred delete engwebs01 rsrctype web user dlucas
  ```

- The following example deletes the resource credential information for the resource printerusers, resource type group, and user name "Mary Jones":

  ```
  pdadmin sec_master> rsrccred delete printerusers rsrctype group
  user "Mary Jones"
  ```

See also

"rsrccred create" on page 93

rsrccred list user

Returns the list of single sign-on credentials for the specified user. The user must exist, or an error is displayed.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrccred list user *user_name*

Options

user_name

Specifies the name of the user for whom the resource credential information applies. The user must exist, or an error is displayed. Examples of user names are dlucas, sec_master and "Mary Jones".

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example returns the list of single sign-on credentials for the specified user dlucas:

```
pdadmin sec_master> rsrccred list user dlucas
```

The output is like:

```
Resource Name: engwebs01
Resource Type: web
```

See also

"rsrccred show" on page 97

rsrccred modify

Changes a single sign-on credential.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrccred modify *resource_name* rsrctype {web|group} set [–rsrcuser *new_resource_userid* [–rsrcpwd *new_resource_password*]] user *user_name*

Options

–rsrcpwd new_resource_password
Specifies the new password for a user at the web server. To change or reset the password information, this optional command must be preceded by a dash (-). Specifying this option without specifying the –rsrcpwd option clears both the resource user ID and the resource password from the resource credential. To set the resource password, you must specify both the resource user ID and the resource password. (Optional)

–rsrcuser new_resource_userid
Specifies the new unique user identification (user ID) for the user at the web server. To change or reset the resource user ID of the user, this optional command must be preceded by a dash (-). (Optional)

A valid new resource user ID is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Examples of user identifications are 4807ws01 and userD4D.

resource_name
Specifies the name that is given to the resource or resource group when the resource was created. The resource or resource group must exist, or an error is displayed.

Examples of resource names are engwebs01 and "Mary Jones Printer".

rsrctype {web|group}
Specifies whether the resource type named is web (resource) or group (resource group) for the single sign-on resource that is associated with the credential. The type of resource must match the resource type that is assigned when the resource or resource group credential was first created.

user *user_name*
Specifies the name of the user for whom the resource credential information applies. The user must exist, or an error is displayed.

Examples of user names are dlucas, sec_master, and "Mary Jones".

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example modifies the password of the user `dlucas` to `newrsrpw` for the specified resource `engwebs01`:

```
pdadmin sec_master> rsrccred modify engwebs01 rsrctype web set
-rsrcuser 4807ws01 -rsrcpwd newrsrpw user dlucas
```

- The following example, entered as one line, modifies the group resource user ID to `user888` for the specified resource `printerusers`:

```
pdadmin sec_master> rsrccred modify printerusers rsrctype group set
-rsrcuser user888 user "Mary Jones"
```

See also

"rsrccred create" on page 93

rsrccred show

Displays the attributes of a single sign-on credential. The credential identifier is composed of a resource name, a resource type, and a user name.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrccred show *resource_name* rsrctype {web|group} user *user_name*

Options

resource_name

Specifies the name of the single sign-on resource or resource group that is associated with the credential. The resource or resource group must exist, or an error is displayed.

Examples of resource names are `engwebs01` and `printerusers`.

rsrctype {web|group}

Specifies whether the resource type named is `web` (resource) or `group` (resource group) for the single sign-on resource that is associated with the credential. The type of resource must match the resource type that is assigned when the resource or resource group was first created.

user *user_name*

Specifies the name of the user that is associated with this credential. The user must exist, or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the specified single sign-on credential:

  ```
  pdadmin sec_master> rsrccred show engwebs01 rsrctype web user dlucas
  ```

  The output is like:

  ```
  Resource Name: engwebs01
  Resource Type: web
  Resource User Id: dlucas
  ```

- The following example displays the specified single sign-on credential:

  ```
  pdadmin sec_master> rsrccred show user888 rsrctype group user "Mary Jones"
  ```

  The output is like:

  ```
  Resource Name: printerusers
  Resource Type: group
  Resource User Id: Mary Jones
  ```

See also

"rsrccred list user" on page 95

rsrcgroup create

Creates and names a resource group.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrcgroup create *resource_group_name* [–desc *description*]

Description

You can use a resource group to represent a set of web servers when the sign-on credential for the set of web servers is the same. In this case, web servers are considered resources.

For example, if the user dlucas has the same identity for web servers engwebs01 and engwebs02, these resources can be added to a resource group called webs4807. Use the **rsrcgroup modify** command to add the resources to the group.

Then, you can create a single sign-on credential for the webs4807 resource group for dlucas. Then, that single sign-on credential can be used to access all the web servers in the webs4807 group.

Options

–desc *description*
Specifies the description to identify this resource group. This parameter is optional. A valid description is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. If the description contains a space, ensure that you enclose the description in double quotation marks. (Optional)

Examples of descriptions are "Engineering Web server – Room 4807" and "Printer in room 345, Bldg 2".

resource_group_name
Specifies the name of the resource group. A valid resource group name is an alphanumeric string that is not case-sensitive. If the resource is a GSO resource, certain characters are not allowed. See "Characters disallowed for GSO names" on page 188 for the list of these characters.

Examples of resource group names are `webs4807`, `engwebs01`, and `IBMprinters`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example creates and names a web resource group `IBMprinters`:

```
pdadmin sec_master> rsrcgroup create IBMprinters
```

- The following example creates and names a web resource group named `webs4807` and provides a description (`"Web servers, Room 4807"`) for that resource:

```
pdadmin sec_master> rsrcgroup create webs4807 -desc "Web servers, Room 4807"
```

See also

"rsrcgroup delete" on page 99

rsrcgroup delete

Deletes a single sign-on resource group.

Requires authentication (administrator ID and password) to use this command.

Syntax

`rsrcgroup delete` *`resource_group_name`*

Options

`resource_group_name`

Specifies the name of the resource group. The resource must exist, or an error is displayed.

Examples of resource group names are `webs4807`, `engwebs01`, and `IBMprinters`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example deletes the named resource group and its associated description information:

```
pdadmin sec_master> rsrcgroup delete webs4807
```

See also

"rsrcgroup create" on page 98

rsrcgroup list

Displays the names of all resource groups that are defined in the user registry.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrcgroup list

Options

None.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example returns a list of all single sign-on resource group names:

```
pdadmin sec_master> rsrcgroup list
```

The output is like:

```
webs4807
websbld3
```

See also

"rsrcgroup show" on page 101

rsrcgroup modify

Adds or removes a single sign-on resource to or from a single sign-on resource group.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrcgroup modify *resource_group_name* add rsrcname *resource_name*

rsrcgroup modify *resource_group_name* remove rsrcname *resource_name*

Options

add rsrcname *resource_name*
Adds a single sign-on resource to the specified single sign-on resource group.

A valid resource name is an alphanumeric string that is not case-sensitive. If the resource is a GSO resource, certain characters are not allowed. See "Characters disallowed for GSO names" on page 188 for the list of these characters.

Examples of resource names are engwebs01 and "Mary Jones Printer".

remove rsrcname *resource_name*

Removes a single sign-on resource from the specified single sign-on resource group.

Examples of resource names are engwebs01 and "Mary Jones Printer".

Note: Depending on the LDAP server in your environment, any attempt to remove a non-existing resource from a group, might generate an error.

resource_group_name

Specifies the name of the resource group to be modified. The resource group must exist, or an error is displayed.

Examples of resource group names are webs4807, engwebs01, and IBMprinters.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example adds the resource named engwebs02 to the existing web resource group webs4807:

```
pdadmin sec_master> rsrcgroup modify webs4807 add rsrcname engwebs02
```

- The following example deletes the resource named engwebs02 from the existing web resource group webs4807:

```
pdadmin sec_master> rsrcgroup modify webs4807 remove rsrcname engwebs02
```

See also

"rsrcgroup create" on page 98

rsrcgroup show

Displays the resource group information for the specified resource group.

Requires authentication (administrator ID and password) to use this command.

Syntax

rsrcgroup show *resource_group_name*

Description

The resource group name, the resource group description, and a list of all resource group members names are displayed. The resource group members are the individual web resources (servers).

Options

resource_group_name

Specifies the name of the resource group.

Examples of resource group names are `webs4807`, `engwebs01`, and `IBMprinters`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example returns the specified single sign-on resource group named `webs4807`:

```
pdadmin sec_master> rsrcgroup show webs4807
```

The output is like:

```
Resource Group Name: webs4807
Description: Web servers, Room 4807
Resource Members:
engwebs01
engwebs02
engwebs03
```

See also

"rsrcgroup list" on page 100

server list

Lists all registered Security Verify Access servers.

Requires authentication (administrator ID and password) to use this command.

Syntax

server list

Description

Lists all registered Security Verify Access servers. The name of the server for all server commands must be entered in the exact format as it is displayed in the output of this command. The **server list** command does not have such a requirement.

Options

None.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example lists registered servers:

```
pdadmin> server list
```

The output is as follows:

```
ivmgrd-master
ivacld-server1
ivacld-server2
```

where `ivmgrd-master` represents the Policy server; `ivacld-server2` and `ivacld-server1` represent Authorization server instances.

server listtasks

Retrieves the list of tasks (commands) available for the specified installed Security Verify Access server or server instance.

Requires authentication (administrator ID and password) to use this command.

Syntax

server listtasks *server_name-host_name*

Options

server_name-host_name

Specifies the name of the installed Security Verify Access server or server instance. You must specify the *server_name* in the exact format as displayed in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server is `default`, the *server_name* is `default-webseald` followed by -*host_name*. The full server name–host name is `default-webseald-cruz.dallas.ibm.com`.

For multiple server instances on the same computer, if the configured name of a WebSEAL server instance is `webseal2-webseald`, the *instance_name* is followed by -*host_name*. The full server instance name–host name is `webseal2-webseald-cruz.dallas.ibm.com`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the list of tasks available from the authorization server:

```
pdadmin sec_master> server listtasks ivacld-mogman.admogman.com
```

The output is like:

```
trace set component level [file path=file|
other-log-agent-config]
trace show [component]
trace list [component]
stats show [component]
stats list
stats on [component] [interval] [count]
[file path= file|other-log-agent-config]
stats off [component]
stats reset [component]
stats get [component]
```

Note: You can run the command `server listtasks ivmgrd-master` to view the list of tasks available from the Policy server. The output is similar to the output listed for the authorization server.

- The following example displays the list of tasks available from the WebSEAL server `default-webseald-cruz`:

```
pdadmin sec_master server listtasks default-webseald-cruz
```

The output is like:

```
dynurl update
jmt load
jmt clear
cache flush all
create
add
remove
delete <junction point>
list
show <junction point>
reload
terminate all_sessions <user_id>
terminate session <user_session_id>
refresh all_sessions <user_id>
help command
trace set component level [file path=file|
other-log-agent-config]
trace show [component]
trace list [component]
stats show [component]
stats list
stats on [component][interval][count]
[file path= file|other-log-agent-config]
stats off [component]
stats reset [component]
stats get [component]
```

See also

“server list” on page 102
“server show” on page 105

server replicate

Notifies the installed Security Verify Access authorization server or server instance to receive database updates.

Requires authentication (administrator ID and password) to use this command.

Syntax

server replicate [–server *server_name-host_name*]

Options

-server ***server_name-host_name***

Specifies the name of the installed Security Verify Access server or server instance. You must specify the *server_name* in the exact format as displayed in the output of the **server list** command. (Optional)

For example, if the configured name of a single WebSEAL server is `default`, the *server_name* is `default-webseald` followed by *-host_name*. The full server name–host name is `default-webseald-cruz.dallas.ibm.com`.

For multiple server instances on the same computer, if the configured name of a WebSEAL server instance is `webseal2-webseald`, the *instance_name* is followed by *-host_name*. The full server instance name–host name is `webseal2-webseald-cruz.dallas.ibm.com`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

Here is an example of this command when the server name is specified:

```
pdadmin sec_master> server replicate -server ivacld-topserver
```

See also

"server list" on page 102
"server show" on page 105

server show

Displays the properties for the specified installed Security Verify Access server or server instance. The server must exist, or an error is displayed.

Requires authentication (administrator ID and password) to use this command.

Syntax

server show *server_name-host_name*

Options

server_name-host_name

Specifies the name of the installed Security Verify Access server or server instance. You must specify the *server_name* in the exact format as displayed in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server is `default`, the *server_name* is `default-webseald` followed by *-host_name*. The full server name–host name is `default-webseald-cruz.dallas.ibm.com`.

For multiple server instances on the same computer, if the configured name of a WebSEAL server instance is `webseal2-webseald`, the *instance_name* is followed by *-host_name*. The full server instance name–host name is `webseal2-webseald-cruz.dallas.ibm.com`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the specified properties for the authorization server `(ivacld)` on the `mogman` computer:

  ```
  pdadmin sec_master> server show ivacld-mogman
  ```

  The output is like:

  ```
  ivacld-mogman
  Description: ivacld/mogman
  Hostname: mogman
  Principal: ivacld/mogman
  Administration Request Port: 7137
  Listening for authorization database update notifications: yes
  AZN Administration Services:
  AZN_ADMIN_SVC_TRACE
  ```

- The following example displays the properties of the WebSEAL server `default-webseald-cruz`:

  ```
  pdadmin sec_master> server show default-webseald-cruz
  ```

  The output is like:

  ```
  default-webseald-cruz
  Description: default-webseald-cruz
  Hostname: cruz.dallas.ibm.com
  Principal: default-webseald/cruz
  Administration Request Port: 7234
  Listening for authorization database update notifications: yes
  AZN Administration Services:
  webseal-admin-svc
  azn_admin_svc_trace
  ```

- The following example displays the `ivmgrd-master` policy server properties:

  ```
  pdadmin sec_master> server show ivmgrd-master
  ```

  The output is like:

  ```
  ivmgrd-master
  Description: ivmgrd-master
  Hostname: localhost
  Principal:
  Administration Request Port: 7135
  Listening for authorization database update notifications: no
  AZN Administration Services:
  azn_admin_svc_trace
  ```

  Note: The Administration Request Port, 7135 in the output, is the same port that is set for the policy server during policy server configuration.

See also

“server list” on page 102
“server task show” on page 130

server task add

Adds an application server to an existing WebSEAL junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* add –h *host_name* [*options*] *junction_point*

Options

–h *host_name*

Specifies the DNS host name or IP address of the target application server. Valid values for *host_name* include any valid IP host name. For example:

```
www.example.com
```

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

junction_point

Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

options

Specifies the options that you can use with the **server task add** command. (Optional) These options include:

–D "*dn*"

Specifies the distinguished name of the server certificate. This value, matched with the actual certificate DN, enhances authentication and provides mutual authentication over SSL. For example, the certificate for `www.example.com` might have the following DN:

```
"CN=WWW.EXAMPLE.COM,OU=Software,O=example.com\, Inc,L=Austin,
ST=Texas,C=US"
```

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–H *host_name*

Specifies the DNS host name or IP address of the proxy server. Valid values for *host_name* include any valid IP host name. For example:

```
www.example.com
```

This option is used for junctions that were created with the type of `tcpproxy` or `sslproxy`.

–i

Indicates that the WebSEAL server does not treat URLs as case-sensitive. This option is used for junctions that were created with the type of `tcp` or `ssl`.

-p *port*
Specifies the TCP port of the server. The default value is 80 for TCP junctions and 443 for SSL junctions. This option is used for junctions that were created with the type of `tcp` or `ssl`.

-P *port*
Specifies the TCP port of the HTTP proxy server. The default value is 7138. Use this option for junctions that were created with the type of `tcpproxy` or `sslproxy`.

For *port*, use any valid port number. A valid port number is any positive number that is allowed by TCP/IP and that is not currently being used by another application. Use the default port number value, or use a port number that is greater than 1000 that is not being used.

This option is also valid for mutual junctions to specify the HTTPS port of the back-end third-party server.

-q *url*
Specifies the relative path for the **query_contents** script. By default, Security Verify Access looks for this script in the `/cgi_bin` subdirectory. If this directory is different or the **query_contents** file is renamed, use this option to indicate to WebSEAL the new URL to the file. Required for Windows servers.

This option is used for junctions that were created with the type of `tcp` or `ssl`.

-u *uuid*
Specifies the UUID of this server when connected to WebSEAL over a stateful junction that was using the `–s` option. This option is used for junctions that were created with the type of `tcp` or `ssl`.

-v *virtual_hostname*
Specifies the virtual host name that is represented on the server. This option supports a virtual host setup on the server. Use this option when the junction server expects a host name header, because you are junctioning to one virtual instance of that server.

The default HTTP header request from the browser does not know that the server has multiple names and multiple virtual servers.

You must configure WebSEAL to supply that extra header information in requests that are destined for a server that is set up as a virtual host.

This option is used for junctions that were created with the type of `tcp` or `ssl`.

-V *virtual_hostname*
Specifies the virtual host name that is represented on the back-end server. This option:

- Supports a virtual host setup on the back-end server.
- Is used only for mutual junctions.
- Corresponds to the virtual host that is used for HTTPS requests.

You can use –V when the back-end junction server expects a host name header and you are junctioning to one virtual instance of that server. The default HTTPS header request from the browser does not know that the back-end server has multiple names and multiple virtual servers. You must configure WebSEAL to supply that extra header information. This header information applies to requests destined for a back-end server that is set up as a virtual host.

-w
Indicates Microsoft Windows 32 bit (Win32) file system support.

This option is used for junctions that were created with the type of `tcp` or `ssl`.

-y priority
The priority for the server (1-9). Default is 9.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`*`/`*`junction_point`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command, and returns an error message.

1

The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about how to add servers to existing junctions, see the Administering topics in the IBM Knowledge Center.

Example

The following example creates a junction for the WebSEAL server named `WS1` to the server named APP1. The example adds another server named APP2 to the same junction point:

```
pdadmin> server task default-webseald-WS1 create -t tcp -h APP1 -s /mnt

pdadmin> server task default-webseald-WS1 add -h APP2 /mnt
```

See also

"server task create" on page 110
"server task delete" on page 117
"server task remove" on page 128
"server task show" on page 130

server task cache flush all

Flushes the HTML document cache.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* `cache flush all`

Options

`instance_name`*`-webseald-`*`host_name`

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`*`/` object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note:

- This command is available only when WebSEAL is installed.
- For more information about the WebSEAL content caching, see the Administering topics in the IBM Knowledge Center.

Example

The following example flushes all web document caches:

```
pdadmin> server task default-webseald-abc.ibm.com cache flush all
```

server task create

Creates a WebSEAL junction point.

Requires authentication (administrator ID and password) to use this command.

Syntax

For local junctions:

`server task` *`instance_name`*`-webseald-`*`host_name`* `create –t` *`type`* `–d` *`dir`* `[`*`options`*`]` *`junction_point`*

For non-local junctions:

`server task` *`instance_name`*`-webseald-`*`host_name`* `create –t` *`type`* `–h` *`host_name`* `[`*`options`*`]` *`junction_point`*

Options

–d *dir*

Specifies the local directory to the junction. This option is required if the junction type is local.

This option is valid only with junctions that were created with the type of `local`.

–h *host_name*

Specifies the DNS host name or IP address of the target server. This option is valid only for non-local junctions; local junctions do not need a host name. Valid values for *host_name* include any valid IP host name. For example:

```
www.example.com
```

–T {*resource* | *resource_group*}

Specifies the name of the resource or resource group. This option is required only when the –b `gso` option is used. This option is valid for all junctions except for the type of `local`.

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical machine where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host machine name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

junction_point

Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

options

Specifies the options that you can use with the **`server task create`** command. (Optional) These options include:

–a *address*

Specifies the local IP address that WebSEAL uses to communicate with the target back-end server. If this option is not provided, WebSEAL uses the default address as determined by the operating system.

If an address is supplied for a particular junction, WebSEAL is modified to bind to this local address for all communication with the junctioned server.

–A

Enables or disables lightweight third-party authentication mechanism (LTPA) junctions. This option requires the –F and –Z options. The –A, –F, and –Z options all must be used together.

This option is valid for all junctions except for the type of `local`.

–2

You can use this option with the –A option to specify that LTPA version 2 cookies (LtpaToken2) are used. The –A option without the –2 option specifies that LTPA version 1 cookies (LtpaToken) are used.

–b *BA_value*

Defines how the WebSEAL server passes the HTTP BA authentication information to the server, which is one of the following values:

- `filter` (default)
- `ignore`
- `supply`
- `gso`

This option is valid for all junctions except for the type of `local`.

–B

Indicates that WebSEAL uses the BA header information to authenticate to the server and to provide mutual authentication over SSL. This option requires the `–U` and `–W` options.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–c *header_type*

Inserts the Security Verify Access client identity in HTTP headers across the junction. The *header_type* argument can include any combination of the Security Verify Access HTTP header types:

- `{iv_user|iv_user_l}`
- `iv_groups`
- `iv_creds`
- `all`

The header types must be comma-separated, and cannot have a space between the types. For example: `-c iv_user,iv_groups`

Specifying `–c all` is the same as specifying `–c iv_user,iv_groups,iv_creds`.

This option is valid for all junctions except for the type of `local`.

–C

Indicates single sign-on from a front-end WebSEAL server to a back-end WebSEAL server. The `–C` option is not mutual authentication.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–D "*dn*"

Specifies the distinguished name of the server certificate. This value, matched with the actual certificate DN, enhances authentication and provides mutual authentication over SSL. For example, the certificate for `www.example.com` might have a DN of

```
"CN=WWW.EXAMPLE.COM,OU=Software,O=example.com\, Inc,L=Austin,
ST=Texas,C=US"
```

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–e *encoding_type*

Specifies the encoding to use when HTTP headers are generated for junctions. This encoding applies to headers that are generated with both the `–c` junction option and tag-value. The following values for encoding are supported:

utf8_bin

WebSEAL sends the headers in UTF-8.

utf8_uri

WebSEAL sends the headers in UTF-8 but URI also encodes them. This behavior is the default behavior.

lcp_bin

WebSEAL sends the headers in the local code page of the WebSEAL server.

lcp_uri

WebSEAL sends the headers in the local code page of the WebSEAL server, but URI also encodes them.

This option is valid for all junctions except for the type of `local`.

–f

Forces the replacement of an existing junction.

This option is used for junctions that were created with any junction type.

–F *keyfile*

Specifies the location of the key file that is used to encrypt LTPA cookie data.

The –F option requires –A and –Z options. The –A, –F, and –Z options all must be used together.

This option is valid for all junctions except for the type of `local`.

–H *host_name*

Specifies the DNS host name or IP address of the proxy server. The –P option also supports proxy server junctions. Valid values for *host_name* include any valid IP host name. For example,

```
proxy.www.example.com
```

This option is valid only with junctions that were created with the type of `tcpproxy` or `sslproxy`.

–i

Indicates that the WebSEAL junction does not treat URLs as case-sensitive. To correctly authorize requests for junctions that are not case-sensitive, WebSEAL does the authorization check on a lowercase version of the URL. For example, a Web server that is running on a Windows operating system treats requests for `INDEX.HTM` and `index.htm` as requests for the same file.

Junctions to such a Web server must be created with the `–i` or `–w` option. ACLs or POPs that are attached to objects beneath the junction point must use the lowercase object name. An ACL attached to `/junction/index.htm` applies to all the following requests if the `–i` or `–w` option is used:

```
/junction/INDEX.HTM
/junction/index.htm
/junction/InDeX.HtM
```

This option is valid for all junctions except for the type of `local`. Local junctions are not case-sensitive only on Win32 platforms; all other platforms are case-sensitive.

–I

Ensures a unique Set-Cookie header name attribute when the `–j` option is used to modify server-relative URLs in requests.

This option is valid for all junctions except for the type of `local`.

–j

Supplies junction identification in a cookie to handle script-generated server-relative URLs.

This option is valid for all junctions except for the type of `local`.

–J trailer,inhead,onfocus,xhtml10

Controls the junction cookie JavaScript block.

Use `–J trailer` to append the junction cookie JavaScript to HTML page returned from back-end server.

Use `–J inhead` to insert the Javascript block between `<head> </head>` tags for HTML 4.01 compliance.

Use `–J onfocus` to use the onfocus event handler in the JavaScript to ensure that the correct junction cookie is used in a multiple-junction/multiple-browser-window scenario.

Use `–J xhtml10` to insert a JavaScript block that is HTML 4.01 and XHTML 1.0 compliant.

Use `–J httpheader` to insert the junction cookie as a standard HTTP cookie in the HTTP response headers.

–k

Sends WebSEAL session cookies to the junction server. By default, cookies are removed from requests that are sent to the server.

This option is valid for all junctions except for the type of `local`.

–K "*key_label*"
Specifies the key label of the client personal certificate that WebSEAL must present to the server. Use of this option allows the junction server to authenticate the WebSEAL server by using client certificates.

This option is valid only with junctions that were created with the type of `ssl` and `sslproxy`.

–l *percent*
Defines the soft limit for consumption of worker threads.

This option is valid for all junctions except for the type of `local`.

–L *percent*
Defines the hard limit for consumption of worker threads.

This option is valid for all junctions except for the type of `local`.

–n
Indicates that no modifications of the names of non-domain cookies are to be made. Use when client side scripts depend on the names of cookies.

WebSEAL modifies the names of non-domain cookies that are returned from the junction to prefix with `AMWEBJCT!`*`junction_point`*. WebSEAL does this action by default, if a junction is listed in the JMT or if the `–j` junction option is used.

This option is valid for all junctions except for the type of `local`.

–p *port*
Specifies the TCP port of the back end third-party server. The default value is 80 for TCP junctions and 443 for SSL junctions.

This option is valid for all junctions except for the type of `local`.

–P *port*
For proxy junctions that were created with the type of `tcpproxy` or `sslproxy` this option specifies the TCP port number for the HTTP proxy server. The –P option is required when the –H option is used.

This option is also valid for mutual junctions to specify the HTTPS port of the back-end third-party server.

–q *path*
Specifies the relative path for the `query_contents` script. By default, Security Verify Access looks for the `query_contents` script in the `/cgi_bin` directory. If this directory is different or the `query_contents` file name is renamed, this option indicates to WebSEAL the new URL to the file. Required for back end Windows servers.

If you want to set Security Verify Access to not get any `query_contents` data from the junctioned server, you can specify this option as `"-q disabled"`.

This option is valid for all junctions except for the type of `local`.

–r
Inserts the incoming IP address into the HTTP header across the junction. This option is valid for all junctions except for the type of `local`.

–R
Allows the request to proceed but provides the rule failure reason to the junction in an HTTP header. If the –R option is not used and a rule failure occurs, WebSEAL does not allow the request to proceed. This option is valid for all junctions except for the type of `local`.

–s
Indicates that the junction support stateful applications. By default, junctions are not stateful. This option is valid for all junctions except for the type of `local`.

–S *path*
Specifies the location of the forms single sign-on configuration file. This option is valid for all junctions except for the type of `local`.

-t *type*
Specifies the type of junction; must be one of the following types:

- `tcp`
- `tcpproxy`
- `ssl`
- `sslproxy`
- `local`

-u *uuid*
Specifies the Universally Unique Identifier (UUID) of a server that is connected to WebSEAL by using a stateful junction (–s option). This option is valid for all junctions except for the type of `local`.

-U "*user_name*"
Specifies the WebSEAL server user name. This option requires the `–B` and `–W` options. WebSEAL uses the BA header information to authenticate to the server and to provide mutual authentication over SSL. This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

-v *virtual_hostname*[:*HTTP-port*]
Specifies the virtual host name for the server. This option supports multiple virtual hosts that are served from the same Web server. Use `–v` when the junction server expects a host name header different from the DNS name of the server. This option is valid for all junctions except for the type of `local`. For mutual junctions, this value corresponds to the virtual host that is used for HTTP requests.

-V *virtual_hostname*[:*HTTPS-port*]
Specifies the virtual host name for the back-end server. This option supports multiple virtual hosts that are served from the same Web server. Use –V when the back-end junction server expects a host name header that is different from the DNS name of the server. This option is used only for mutual junctions and corresponds to the virtual host that is used for HTTPS requests.

-w
Indicates Microsoft Windows 32-bit (Win32) file system support. This option:

- Provides all the functionality that is provided by the `–i` junction option.
- Disallows requests that contain file names that might be interpreted as Win32 file name aliases.

The option is valid for all junctions except for the type of `local`. Local junctions prohibit URLs that contain Win32 file name aliases on Win32 but allow such URLs on other platforms.

-W "*password*"
Specifies the WebSEAL server password. This option requires the –B and –U options. WebSEAL uses the BA header information to authenticate to the server and to provide mutual authentication over SSL. This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

-x
Creates a path junction that is not apparent.

This option is valid for all junctions except for the type of `local`.

-y priority
The priority for the server (1-9). Default is 9.

-Y
Enables the Federation Runtime single sign-on (SSO) for the junction.

Indicates that Kerberos SSO is enabled for the junction. Before you use this command, configure the WebSEAL configuration file to support Kerberos single sign-on over junctions.

–Z *keyfile_pwd*

Specifies the password of the key file that is used to encrypt LTPA cookie data. This option requires the –A and –F options. The –A, –F, and –Z options all must be used together. This option is valid for all junctions except for the type of `local`.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`*`/`*`junction_point`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note:

- This command is available only when WebSEAL is installed.
- For more information about creating a junctioned server, see the Administering topics in the IBM Knowledge Center.
- For more information about gathering statistics, see the Auditing topics in the IBM Knowledge Center.

Examples

- The following example creates a basic WebSEAL junction `/pubs` on the `default-webseald-cruz` WebSEAL server. The junction type is TCP, and the host name is `doc.tivoli.com`:

  ```
  pdadmin> server task default-webseald-cruz create -t tcp \
  -h doc.tivoli.com /pubs
  ```

  Output is like:

  ```
  Created junction at /pubs
  ```

- The following example creates a new local junction / to replace the current junction point. The `–f` option is required to force a new junction that overwrites an existing junction at the `/tmp/docs` directory:

  ```
  pdadmin> server task default-webseald-cruz create -t local \
  -f -d /tmp/docs /
  ```

  Output is like:

  ```
  Created junction at /
  ```

- The following example limits worker thread consumption on a per junction basis with a:
  - Soft thread limit of `60`.
  - Hard thread limit of `80`.

  The junction in this example is `/myjunction`.

  ```
  pdadmin> server task default-webseald-cruz create -t tcp \
  -h cruz.dallas.ibm.com -l 60 -L 80 /myjunction
  ```

See also

“server task add” on page 107
“server task delete” on page 117
“server task remove” on page 128
“server task show” on page 130

server task delete

Deletes a WebSEAL junction point.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* delete *junction_point*

Options

instance_name*-webseald-*host_name
: Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

junction_point
: Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/*junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
: The command completed successfully. For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
: The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note:

- This command is available only when WebSEAL is installed.
- For more information about how to delete WebSEAL junctions, see the Administering topics in the IBM Knowledge Center.

Example

The following example deletes the junction point /pubs from the WebSEAL server `default-webseald-abc.ibm.com`:

```
pdadmin> server task default-webseald-abc.ibm.com delete /pubs
```

See also

“server task add” on page 107
“server task create” on page 110
“server task remove” on page 128
“server task show” on page 130

server task dynurl update

Reloads the dynamic URL configuration file.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* dynurl update

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/ object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about how WebSEAL handles dynamic URLs, see the Administering topics in the IBM Knowledge Center.

Example

The following example reloads the dynamic URL configuration file:

```
pdadmin> server task default-webseald-abc.ibm.com dynurl update
```

server task help

Lists detailed help information about a specific **server task** command.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* help *task*

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

task

Lists detailed help for the specified task, such as the command syntax, the description, and the valid options.

Authorization

No special authorization required.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays output after help is requested for the **server task add** command at the abc.ibm.com WebSEAL server:

  ```
  pdadmin> server task default-webseald-abc.ibm.com help add
  ```

  Output is like:

  ```
  Command:
  add <options> <junction point>
  Description:
  Adds an additional server to a junction
  Usage:
  TCP and SSL Junction Flags
  -i Server treats URLs as case insensitive.
  -h <hostname> Target host (required flag).
  -p <port> TCP port of server.
  Default is 80 for TCP junctions
  443 for SSL junctions.
  -H <hostname> Proxy hostname.
  -P <port> Port of proxy server.
  -D <"DN"> The Distinguished Name of the server
  -q <relative url> URL for query_contents script.
  -u <UUID> (stateful junctions only).
  -v <hostname> Virtual hostname for server.
  -w Win32 file system support.
  -j Scripting support for junction.
  Common Flags
  <junction point> Where to create the junction
  ```

- The following example displays the output after help is requested for the **server task create** command at the abc.ibm.com WebSEAL server:

  ```
  pdadmin> server task default-webseald-abc.ibm.com help create
  ```

  Output is like:

  ```
  Command:
  create -t <type> <options> <junction point>
  Description:
  Creates a new junction
  Usage:
  create -t <type> <options> <junction point>

  TCP and SSL Junction Flags
  ...
  Common Flags
  -t <type>Type of junction.
  One of: tcp, tcpproxy, ssl, sslproxy, local.
  -f Force the creation: overwrite existing junction.
  -R WebSEAL will send the Boolean Rule Header to these
  junctions when a rule failure reason is provided.
  <junction point> Where to create the junction
  ```

See also

"help" on page 53

server task jmt

Clears or loads the junction mapping table data.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* jmt load

server task *instance_name*-webseald-*host_name* jmt clear

Options

`clear`
Clears the junction mapping table data.

`instance_name`**`-webseald-`*****`host_name`***
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named `web2`, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

`load`
Loads the junction mapping table data, which is in the `jmt.conf` file. This file does not exist by default, so you must create the file and add data.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/` object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully.

Note: For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about the WebSEAL junction mapping table, see the Administering topics in the IBM Knowledge Center.

Example

The following example loads the junction mapping table data from the `jmt.conf` file. As a result, WebSEAL has the new information:

```
pdadmin> server task default-webseald-abc.ibm.com jmt load
```

Output is like:

```
JMT table successfully loaded.
```

See also

"server task reload" on page 127

server task list

Lists all junction points on a WebSEAL server or server instance.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name-webseald-host_name* list

Options

instance_name-webseald-host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, if the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

Authorization

Users and groups that require access to this command must be given the **l** (list) permission in the ACL that governs the /WebSEAL/*host_name-instance_name*/*per_junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command, and returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about WebSEAL junctions, see the Administering topics in the IBM Knowledge Center.

Example

The following example lists all junction points on the `default-webseald-cruz` WebSEAL server:

```
pdadmin> server task default-webseald-cruz list
```

Output is like:

```
/
/ssljct
/tcpjct
```

See also

“server task add” on page 107
“server task create” on page 110
“server task delete” on page 117
“server task remove” on page 128
“server task show” on page 130

server task offline

Places the server that is at this junction in an offline operational state.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* `offline` [`–i` *`server_uuid`*] *`junction_point`*

Description

The **`server task offline`** command places the server that is at this junction in an offline operational state. No additional requests are sent to the specified server. If a server is not specified, all servers that are at this junction are placed in an offline operational state.

Options

–i *server_uuid*
Specifies the UUID of the server to place in an offline operational state. If a server is not specified, all servers that are at this junction are placed in an offline operational state. Use the **`server task show`** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

junction_point
Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name-instance_name*/*junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example places the backappl server at the /pubs junction point in an offline operational state. To determine the UUID of this junctioned server, run the **server task show** command:

```
pdadmin> server task default-webseald-cruz show /pubs
```

Output is like:

```
Junction point: /pubs
...
Server 1:
ID: 6fc3187a-ea1c-11d7-8f4e-09267e38aa77
Server State: running
Operational State: Throttled
Throttled at: 2005-03-01-17:07:24
Hostname: backapp1.diamond.example.com
...
Current requests: 0
...
```

Place this server in an offline operational state:

```
pdadmin> server task default-webseald-cruz offline \
-i 6fc3187a-ea1c-11d7-8f4e-09267e38aa77 /pubs
```

See also

“server task online” on page 125
“server task throttle” on page 149
“server task virtualhost offline” on page 163
“server task virtualhost online” on page 165
“server task virtualhost throttle” on page 170

server task online

Places the server that is at this junction in an online operational state.

Syntax

server task *instance_name*-webseald-*host_name* online [–i *server_uuid*] *junction_point*

Description

The **server task online** command places the server that is at this junction in an online operational state. The server now resumes normal operation. If a server is not specified, all servers that are at this junction are placed in an online operational state.

Options

–i *server_uuid*
Specifies the UUID of the server to place in an online operational state. If a server is not specified, all servers that are at this junction are placed in an online operational state. Use the **server task show** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

junction_point
Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/*junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example places the backappl server at the /pubs junction point in an online operational state. To determine the UUID of this junctioned server, run the **server task show** command:

```
pdadmin> server task default-webseald-cruz show /pubs
```

Output is like:

```
Junction point: /pubs
...
Server 1:
ID: 6fc3187a-ea1c-11d7-8f4e-09267e38aa77
Server State: running
Operational State: Offline
Hostname: backapp1.diamond.example.com
...
Current requests: 0
...
```

Place this server in an online operational state:

```
pdadmin> server task default-webseald-cruz online \
-i 6fc3187a-ea1c-11d7-8f4e-09267e38aa77 /pubs
```

See also

"server task offline" on page 123
"server task throttle" on page 149
"server task virtualhost offline" on page 163
"server task virtualhost online" on page 165
"server task virtualhost throttle" on page 170

server task refresh all_sessions

Refreshes the credential for all sessions for a specified user.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* refresh all_sessions *user_id*

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

user_id

Refreshes the credential for all sessions that are associated with the specified user. Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/` object. For example, the **sec_master** administrative user is given this permission by default.

Note: This command is available only when WebSEAL is installed.

Return codes

0

The command completed successfully.

Note: For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: For more information about the WebSEAL credential refresh, see the Administering topics in the IBM Knowledge Center.

Example

The following example refreshes all sessions for the `test_user` user:

```
pdadmin> server task default-webseald-cruz refresh all_sessions test_user
```

See also

"server task terminate session" on page 148
"server task terminate all_sessions" on page 147

server task reload

Reloads the junction mapping table from the database.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* `reload`

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named `web2`, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*host_name*`-`*instance_name*`/` object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about the WebSEAL junction mapping table, see the Administering topics in the IBM Knowledge Center.

Example

The following example reloads the junction mapping table from the database:

```
pdadmin> server task default-webseald-abc.ibm.com reload
```

See also

"server task jmt" on page 120

server task remove

Removes the specified installed WebSEAL server or server instance from a WebSEAL junction point.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *instance_name*-webseald-*host_name* remove –i *server_uuid junction_point*

Options

-i *server_uuid*

Specifies the UUID of the server to be removed from the junction point. See the **`server task show`** command for details about obtaining the UUID.

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

junction_point

Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/*junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: For more information about how to remove a server from a WebSEAL junction, see the Administering topics in the IBM Knowledge Center.

This command is available only when WebSEAL is installed.

Example

The following example removes the `backappl` junctioned server from the `/pubs` junction point. To determine the UUID of the server to be removed, run the **`server task show`** command:

```
pdadmin> server task default-webseald-cruz show /pubs
```

Output is like:

```
Junction point: /pubs
...
Server 1:
ID: 6fc3187a-ea1c-11d7-8f4e-09267e38aa77
Server State: running
...
Hostname: backapp1.cruz.ibm.com
...
```

Remove the server from the junction:

```
pdadmin> server task default-webseald-cruz remove \
-i 6fc3187a-ea1c-11d7-8f4e-09267e38aa77 /pubs
```

See also

"server task add" on page 107
"server task create" on page 110
"server task delete" on page 117
"server task show" on page 130

server task show

Displays detailed information about the specified WebSEAL junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* show *`junction_point`*

Options

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named `web2`, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

junction_point
Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **l** (list) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/`*`junction_point`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about WebSEAL junctions, see the Administering topics in the IBM Knowledge Center.

Example

The following example shows information for the local root junction point / on the WebSEAL server `abc.ibm.com`:

```
pdadmin> server task default-webseald-abc.ibm.com show
```

Output is like:

```
Junction point: /
Type: Local
Junction hard limit: 0 - using global value
Junction soft limit: 0 - using global value
Active worker threads: 0
Root Directory: /opt/pdweb/www-default/docs
...
Server 1:
ID: 78a1eb8c-074a-11d9-abda-00096bda9439
...
```

See also

"server task add" on page 107
"server task create" on page 110
"server task delete" on page 117
"server task remove" on page 128

server task sms key change

Forces the creation of a new session management key.

You might want to forcibly create a key when you suspect that the existing key was compromised.

Syntax

`server task` *`server_name-host_name`* sms key change

Options

`server_name-host_name`

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **`server list`** command.

For example, if the configured name of a single WebSEAL server on host `cruz.dallas.ibm.com` is `default`, the *`server_name`* would be `default-webseald` and the *`host_name`* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If there are multiple configured server instances on the same computer, for example, the host `cruz.dallas.ibm.com`, and the configured name of the WebSEAL server instance is `webseal2-webseald`, the *`server_name`* is `webseal2-webseald` and the *`host_name`* is `example.dallas.ibm.com`. For this example, the name of the server instance would be `webseal2-webseald-example.dallas.ibm.com`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example forcibly creates a session management key for the `abc.ibm.com` server:

```
pdadmin> server task default-webseald-abc.ibm.com key change
```

See also

“server list” on page 102
“server task sms key show” on page 132

server task sms key show

Lists detailed information about the current session management key.

Syntax

server task *server_name–host_name* sms key show

Options

server_name–host_name
Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *server_name* would be `default-webseald` and the *host_name* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same machine, for example:

- The host is `cruz.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *server_name* would be `webseal2-webseald`.
- The *host_name* would be `example.dallas.ibm.com`.
- The name of the server instance would be `webseal2-webseald-example.dallas.ibm.com`.

Return codes

0
The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example returns detailed information about the current session management key for the `abc.ibm.com` server:

```
pdadmin> server task default-webseald-abc.ibm.com sms key show
```

Output is like:

```
ID: 1
Created: 2004-03-03-09:00:03
Expires: 2004-09-03-09:00:03
```

See also

"server list" on page 102
"server task sms key change" on page 131

server task sms realm list

Lists all session management realms in the domain.

Syntax

server task *server_name*–*host_name* sms realm list

Options

server_name–host_name

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *server_name* would be `default-webseald` and the *host_name* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If there are multiple configured server instances on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *server_name* is `webseal2-webseald`.
- The *host_name* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

Return codes

0

The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example lists the realms for the `abc.ibm.com` server:

```
pdadmin> server task default-webseald-abc.ibm.com sms realm list
```

See also

“server list” on page 102
“server task sms realm show” on page 134
“server task sms replica set list” on page 137
“server task sms replica set show” on page 138

server task sms realm show

Lists all replica sets in the specified session management realm.

Syntax

server task *server_name–host_name* sms realm show *realm_name*

Options

realm_name
Specifies the name of the realm. When you specify a realm, the output contains only those replica sets in that realm.

server_name–host_name
Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *server_name* would be `default-webseald` and the *host_name* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *server_name* is `webseal2-webseald`.
- The *host_name* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

Return codes

0
The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example returns the replica sets in the `ibm.com` realm of the `abc.ibm.com` server:

```
pdadmin> server task default-webseald-abc.ibm.com sms realm show ibm.com
```

See also

"server list" on page 102
"server task sms realm list" on page 133
"server task sms replica set list" on page 137
"server task sms replica set show" on page 138

server task sms session refresh all_sessions

Refreshes the credential for sessions for a specific user.

Syntax

`server task` *`server_name–host_name`* sms session refresh all_sessions *`user_name`* –realm *`realm_name`*

Options

`–realm` ***`realm_name`***

Specifies that name of the realm. The credentials of only those sessions that belong to the specified realm are refreshed.

`server_name–host_name`

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **`server list`** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *`server_name`* would be `default-webseald` and the *`host_name`* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is`example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *`server_name`* is `webseal2-webseald`.
- The *`host_name`* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

`user_name`

Refreshes the credential for all sessions that are associated with the specified user. Examples of user names are `dlucas`, `sec_master`, and "`Mary Jones`".

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example refreshes all sessions for user `johnq` in the `ibm.com` realm:

```
pdadmin> server task default-webseald-cruz sms session refresh all_sessions johnq \
-realm ibm.com
```

See also

"server task sms session terminate session" on page 141
"server task sms session terminate all_sessions" on page 140

server task sms session refresh session

Refreshes the credential for a session.

Syntax

`server task` *`server_name–host_name`* `sms session refresh session` *`session_id`* `–realm` *`realm_name`*

Options

`-realm` ***`realm_name`***

Specifies that name of the realm. The credentials of only those sessions that belong to the specified realm are refreshed.

`server_name–host_name`

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **`server list`** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *`server_name`* would be `default-webseald` and the *`host_name`* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *`server_name`* is `webseal2-webseald`.
- The *`host_name`* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

session_id
Specifies the identifier for the session to refresh.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example refreshes session 678 in the `ibm.com` realm:

```
pdadmin> server task default-webseald-cruz sms session refresh session 678 \
-realm ibm.com
```

See also

"server task sms session terminate session" on page 141
"server task sms session terminate all_sessions" on page 140

server task sms replica set list

Lists all session management replica sets in the domain.

Syntax

server task *server_name-host_name* sms replica set list [-realm *realm_name*]

Options

-realm *realm_name*
Indicates that the returned list of replica sets is limited to those replica sets in the specified realm. (Optional)

server_name-host_name
Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *server_name* would be `default-webseald` and the *host_name* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *server_name* is `webseal2-webseald`.
- The *host_name* is `example.dallas.ibm.com`.

- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

Return codes

0
: The command completed successfully.

1
: The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example lists the replica sets in the ibm realm of the abc.ibm.com server:

```
pdadmin> server task default-webseald-abc.ibm.com sms replica set list -realm ibm
```

See also

"server list" on page 102
"server task sms realm list" on page 133
"server task sms realm show" on page 134
"server task sms replica set show" on page 138

server task sms replica set show

Lists all session management replicas in the specified replica set with the time and date that each joined the realm.

Syntax

server task *server_name-host_name* sms replica set show *replica_set_name*

Options

replica_set_name
: Specifies the name of the replica set.

server_name-host_name
: Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is example.dallas.ibm.com.
- The configured name of the WebSEAL server instance is webseal2-webseald.

Then,

- The *server_name* is webseal2-webseald.
- The *host_name* is example.dallas.ibm.com.

- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example returns details about the `ibm.com` replica that is set of the `abc.ibm.com` server:

```
pdadmin> server task default-webseald-abc.ibm.com sms replica set show ibm.com
```

See also

"server list" on page 102
"server task sms realm list" on page 133
"server task sms realm show" on page 134
"server task sms replica set list" on page 137

server task sms session list

Lists all session management sessions.

Syntax

`server task` *`server_name-host_name`* `sms session list –realm` *`realm_name pattern maximum_return`*

Options

`-realm` ***`realm_name`***

Specifies the name of the session management realm.

`server_name–host_name`

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **`server list`** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *`server_name`* would be `default-webseald` and the *`host_name`* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *`server_name`* is `webseal2-webseald`.
- The *`host_name`* is `example.dallas.ibm.com`.

- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

maximum_return
Specifies the maximum number of sessions to return. When there are more matches than designated by this option, the output contains the number of matches.

pattern
Specifies the pattern for returning user names. The pattern can include a combination of wildcard and string constant characters. The pattern is case-sensitive. For example, you can specify *luca* as the pattern to find all users that contain the substring luca within the user name.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example is entered as one line. The example:

- Lists the user sessions in the ibm.com realm of the abc.ibm.com server.
- Lists sessions for users that contain the string ons.
- Limits the number of matches to 100.

```
pdadmin> server task default-webseald-abc.ibm.com
sms session list -realm ibm.com *ons* 100
```

See also

"server list" on page 102
"server task sms realm list" on page 133
"server task sms realm show" on page 134
"server task sms replica set show" on page 138

server task sms session terminate all_sessions

Terminates all user sessions for a specific user.

Syntax

server task *server_name-host_name* sms session terminate all_sessions *user_id* -realm *realm_name*

Options

-realm *realm_name*
Specifies that name of the session management realm.

server_name-host_name
Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host `example.dallas.ibm.com` is `default`, the *server_name* would be `default-webseald` and the *host_name* would be `example.dallas.ibm.com`. For this example, the name of the server would be `default-webseald-example.dallas.ibm.com`.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *server_name* is `webseal2-webseald`.
- The *host_name* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

user_id

Specifies the name of the user. Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example terminates all sessions for the `dlucas` user in the `ibm.com` realm of the `default-webseald-cruz` WebSEAL server:

```
pdadmin> server task default-webseald-cruz sms session terminate \
 all_sessions dlucas -realm ibm.com
```

See also

"server task sms session refresh session" on page 136
"server task sms session refresh all_sessions" on page 135
"server task sms session terminate session" on page 141

server task sms session terminate session

Terminates a user session by using a session ID.

Syntax

server task *server_name–host_name* sms session terminate session *session_id* –realm *realm_name*

Options

server_name–host_name

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is example.dallas.ibm.com.
- The configured name of the WebSEAL server instance is webseal2-webseald.

Then,

- The *server_name* is webseal2-webseald.
- The *host_name* is example.dallas.ibm.com.
- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

session_id

Specifies the ID of a user session.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example terminates session 678 in the ibm.com realm of the default-webseald-cruz WebSEAL server:

```
pdadmin> server task default-webseald-cruz sms session terminate \
session 678 -realm ibm.com
```

See also

"server task sms session refresh all_sessions" on page 135
"server task sms session terminate all_sessions" on page 140

server task sms trace get

Displays the trace level for the session management server.

Syntax

server task *server_name–host_name* sms trace get

Options

server_name–host_name

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is example.dallas.ibm.com.
- The configured name of the WebSEAL server instance is webseal2-webseald.

Then,

- The *server_name* is webseal2-webseald.
- The *host_name* is example.dallas.ibm.com.
- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example returns the tracing level for the ivacld-cruz authorization server:

```
pdadmin> server task ivacld-cruz.dallas.ibm.com sms trace get
```

See also

"server task sms trace set" on page 143

server task sms trace set

Sets the trace level for the distributed session cache.

Syntax

server task *server_name–host_name* sms trace set *level*

Options

level

Specifies the level of tracing. A valid setting is an integer between 0 and 3, with 3 being the most detailed level of trace.

server_name-host_name

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is example.dallas.ibm.com.
- The configured name of the WebSEAL server instance is webseal2-webseald.

Then,

- The *server_name* is webseal2-webseald.
- The *host_name* is example.dallas.ibm.com.
- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when the session management command-line extensions are installed to a hosting authorization server.

Example

The following example sets the tracing level to 1 on the ivacld-cruz authorization server:

```
pdadmin> server task ivacld-cruz.dallas.ibm.com sms trace set 1
```

See also

"server task sms trace get" on page 142

server task stats

Manages the gathering and reporting of statistics for Security Verify Access servers and server instances.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *server_name–host_name* stats get [*component*]

server task *server_name–host_name* stats list

server task *server_name–host_name* stats off [*component*]

server task *server_name–host_name* stats on *component* [*interval* [*count*]] [*destination*]

server task *server_name–host_name* stats reset [*component*]

server task *server_name–host_name* stats show [*component*]

Description

The **server task stats** command manages the gathering and reporting of statistics for Security Verify Access servers and server instances. You can use the **stats** commands with configuration setting that are defined by the stanza entries in the server configuration file to manage statistics.

Statistics gathering is enabled through:

- The **stats on** command.
- The defined configuration settings.

Then, you can use the **stats on** commands to modify the behavior for gathering and reporting statistics.

For example, statistics are enabled to create five statistics reports with each report generated each day. You can use the **stats on** command to change the frequency to every 12 hours. For this example, assume that the following command started statistics gathering:

```
pdadmin sec_master> server task PDWebPI-linuxweb.wasp.ibm.com stats on \
pdwebpi.stats 86400 5 file path=/tmp/stats.log
```

To modify the interval to 12 hours and create 10 reports, issue the following command:

```
pdadmin sec_master> server task PDWebPI-linuxweb.wasp.ibm.com stats on \
pdwebpi.stats 43200 10
```

Although the destination is not specified, the statistics infrastructure assumes any preexisting value. Entering the previous command does disable statistics from being written to the previously defined log file. However, if you specified a different destination, statistics reports would be written to the new destination only. You cannot use the **stats on** command to write statistics reports to more than one destination.

For more information about gathering statistics, see the Auditing topics in the IBM Knowledge Center.

Options

component
Specifies the component about which to gather or report statistics.

count
Specifies the number of reports to send to a log file. When you use the *count* option, you must specify the *interval* option. If you specify the *interval* option without the *count* option, the duration of reporting is indefinite.

After the count value is reached, reporting to a log file stops. Although statistics are no longer sent to a log file, the statistic component is still enabled. You can obtain reports from memory by using the **stats get** command.

destination
Specifies where the gathered statistics are written, where *destination* can be one of the following options:

file path=*file_name*
Specifies the fully qualified name of the log file.

log_agent
Specifies a directory where statistics information is gathered. For more information about logging events, see the Troubleshooting topics in the IBM Knowledge Center.

get
Displays the current report for a specific component or for all enabled components. If you specify the *component* option, displays the current report for that component; otherwise, displays the current report for all enabled components.

interval
Specifies the interval in seconds when statistics are sent from memory to a log file. When this option is specified, statistics are sent, by default, to the server-specific log file designated by the logcfg

entry in the server configuration file. You can specify another location by using the *destination* option. If an interval is not specified, statistics are not sent to a log file, but remain in memory.

Although statistics are not sent to a log file, the statistic component is still enabled. You can obtain reports from memory by using the **stats get** command.

list
Lists all components that are available to gather and report statistics.

off
Disables gathering of statistics for a specific component or for all components. If you specify the *component* option, disables gathering of statistics for that component; otherwise, disables gathering of statistics for all components.

on
Enables gathering of statistics for a specific component. When you enable gathering of statistics, you can also set the reporting frequency, count, and log file.

reset
Resets gathering of statistics for a specific component or for all enabled components. If you specify the *component* option, resets gathering of statistics for that component; otherwise, resets gathering of statistics for all components.

server_name–host_name
Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is example.dallas.ibm.com.
- The configured name of the WebSEAL server instance is webseal2-webseald.

Then,

- The *server_name* is webseal2-webseald.
- The *host_name* is example.dallas.ibm.com.
- The name of the server instance is webseal2-webseald-example.dallas.ibm.com.

show
Lists all enabled components or indicates whether a specific component is enabled. If you specify the *component* option and the component is enabled, the output lists that component; otherwise, no output is displayed. If you do not specify the *component* option, the output lists all enabled components.

Return codes

0
The command completed successfully.

1
The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example uses the **stats list** command to lists all enabled components on the ivacld-mogman.admogman.com authorization server:

```
#pdadmin sec_master> server task ivacld-mogman.admogman.com stats list
```

```
pd.ras.stats.monitor
pd.log.EventPool.queue
```

- The following example:
  - Uses the **stats on** command to enable gathering of statistics for the pd.log.EventPool.queue component on the ivacld-mogman.admogman.com authorization server.
  - Sets the reporting frequency to 30 days, that is, 2592000 seconds.
  - Sets the destination to the c:\myEPstats.log log file.

```
#pdadmin sec_master> server task ivacld-mogman.admogman.com stats on \
pd.log.EventPool.queue 2592000 file path=c:\myEPstats.log
```

See also

"server list" on page 102

server task terminate all_sessions

Terminates all user sessions for a specific user.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* terminate all_sessions *user_id*

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example,

- The configured name of a single WebSEAL instance is default.
- The host computer name that has the WebSEAL server that is installed is abc.ibm.com.

Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

user_id

Specifies the name of the user. Examples of user names are dlucas, sec_master, and "Mary Jones".

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/ object. For example, the **sec_master** administrative user is given this permission by default.

Note: This command is available only when WebSEAL is installed.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: For more information about the WebSEAL server tasks and junction points, see the Administering topics in the IBM Knowledge Center.

Example

The following example terminates all sessions for the dlucas user on the default-webseald-cruz WebSEAL server:

```
pdadmin> server task default-webseald-cruz terminate all_sessions dlucas
```

See also

"server task terminate session" on page 148
"server task refresh all_sessions" on page 126

server task terminate session

Terminates a user session by using a session ID.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* terminate session *session_id*

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example:

- The configured name of a single WebSEAL instance is default.
- The host computer name where the WebSEAL server is installed is abc.ibm.com.

Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

session_id
Specifies the ID of a user session.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the /WebSEAL/*host_name-instance_name*/ object. For example, the **sec_master** administrative user is given this permission by default.

Note: This command is available only when WebSEAL is installed.

Return codes

0
The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: For more information about the WebSEAL server tasks and junction points, see the Administering topics in the IBM Knowledge Center.

Example

The following example (entered as one line) terminates a specific session on the default-webseald-cruz WebSEAL server:

```
pdadmin> server task default-webseald-cruz terminate
session 6fc3187a-ea1c-11d7-8f4e-09267e38aa77
```

See also

"server task refresh all_sessions" on page 126
"server task terminate all_sessions" on page 147

server task throttle

Places the server that is at this junction in a throttled operational state.

Syntax

server task *instance_name*-webseald-*host_name* throttle [–i *server_uuid*] *junction_point*

Description

The **server task throttle** command places the server that is at this junction in a throttled operational state. Users can create a session with WebSEAL before the invocation of this command. Only requests from such users are processed by the specified server. If a server is not specified, all servers that are at this junction are placed in a throttled operational state.

Options

***-i** server_uuid*

Specifies the UUID of the server to throttle. If a server is not specified, all servers that are at this junction are placed in a throttled operational state. Use the **server task show** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example:

- The configured name of a single WebSEAL server instance is default.
- The host computer name where the WebSEAL server is installed is abc.ibm.com.

Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

junction_point

Specifies the name of the directory in the WebSEAL protected object space where the document space of the server is mounted.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/*junction_point* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example places the backappl server that is located at the /pubs junction point in a throttled operational state. To determine the UUID of this junctioned server, run the **server task show** command:

```
pdadmin> server task default-webseald-cruz show /pubs
```

Output is like:

```
Junction point: /pubs
...
Server 1:
ID: 6fc3187a-ea1c-11d7-8f4e-09267e38aa77
Server State: running
Operational State: Online
Hostname: backapp1.diamond.example.com
...
Current requests: 0
...
```

Place this server in a throttled operational state:

```
pdadmin> server task default-webseald-cruz throttle \
-i 6fc3187a-ea1c-11d7-8f4e-09267e38aa77 /pubs
```

See also

"server task offline" on page 123
"server task online" on page 125
"server task virtualhost offline" on page 163
"server task virtualhost online" on page 165
"server task virtualhost throttle" on page 170

server task trace

Enables the gathering of trace information for components of installed Security Verify Access servers or server instances.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *server_name–host_name* trace list [*component*]

server task *server_name–host_name* trace set *component level* [*destination*]

server task *server_name–host_name* trace show [*component*]

Description

The **server task stats** command enables the gathering of trace information for components of installed Security Verify Access servers or server instances that support debug event tracing. The content of trace messages is undocumented and is intended to be used for debugging purposes only. The format and content of trace messages might vary between product releases.

Options

list [*component*]

Lists all enabled trace components that are available to gather and report trace information. If you specify the *component* option and the component is enabled, the output lists that component; otherwise, no output is displayed. If you do not specify the *component* option, the output lists all enabled components.

server_name–host_name

Specifies the name of the server or server instance. You must specify the server name in the exact format as it is shown in the output of the **server list** command.

For example, if the configured name of a single WebSEAL server on host example.dallas.ibm.com is default, the *server_name* would be default-webseald and the *host_name* would be example.dallas.ibm.com. For this example, the name of the server would be default-webseald-example.dallas.ibm.com.

If multiple server instances are configured on the same computer, for example:

- The host is `example.dallas.ibm.com`.
- The configured name of the WebSEAL server instance is `webseal2-webseald`.

Then,

- The *`server_name`* is `webseal2-webseald`.
- The *`host_name`* is `example.dallas.ibm.com`.
- The name of the server instance is `webseal2-webseald-example.dallas.ibm.com`.

set *component level* [*destination*]

Sets the trace level and trace message destination for a specific *component* and its subordinates. The value for the *level* option is a single integer from 1 to 9, with 9 reporting the most detailed level of information. The *destination* option specifies where the gathered trace information is written and can be one of the following options:

file path=*file_name*

Specifies the fully qualified file name.

log_agent

Specifies a destination for the statistics information that is gathered for the specified component. For more information about logging events, see the Administering topics in the IBM Knowledge Center.

show [*component*]

Shows the names and levels for all enabled trace components. If you specify the *component* option, the output lists the name and level for the specified component.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example enables the `pdweb.debug` trace component to level 2 and then displays the output for all enabled components. WebSEAL–specific components are prefixed with pdweb.

```
pdadmin sec_master> server task webseald-instance_name trace set
pdweb.debug 2

pdadmin sec_master> server task webseald-instance_name trace show
```

Output from the **trace show** command is like:

```
pdweb.debug 2
```

- The following example enables the `pdwebpi.module.session-cookie` trace component to level 9. Then, the output for all enabled components is displayed. Components that are specific to the web server plug-ins are prefixed with `pdwebpi`.

```
pdadmin sec_master> server task pdwpi-tivoli.com trace set
pdwebpi.module.session-cookie 9

pdadmin sec_master> server task pdwpi-tivoli.com trace show
```

Output from the **trace show** command is like:

```
pdwebpi.module.session-cookie 9
```

See also

"server list" on page 102

server task virtualhost add

Adds an additional installed WebSEAL server or server instance to an existing virtual host junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost add –h *host_name* [*options*] *vhost_label*

Options

–h *host_name*
Specifies the DNS host name or IP address of the target server. Valid values for *host_name* include any valid IP host name. This option is required. For example:

```
www.example.com
```

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

options
Specifies the options that you can use with the **server task virtualhost add** command. (Optional) These options include:

–D "*dn*"
Specifies the distinguished name of the server certificate. This value, matched with the actual certificate DN, enhances authentication and provides mutual authentication over SSL. For example, the certificate for www.example.com might have a DN of

```
"CN=WWW.EXAMPLE.COM,OU=Software,O=example.com\, Inc,L=Austin,
ST=Texas,C=US"
```

This option is valid only with junctions that were created with the type of ssl or sslproxy.

–H *host_name*
Specifies the DNS host name or IP address of the proxy server.

Valid values for *host_name* include any valid IP host name. For example:

```
proxy.www.example.com
```

This option is used for junctions that were created with the type of `tcpproxy` or `sslproxy`.

-i

Indicates that the WebSEAL server does not treat URLs as case-sensitive.

This option is used for junctions that were created with the type of `tcp` or `ssl`.

-p *port*

Specifies the TCP port of the server. The default value is 80 for TCP junctions and 443 for SSL junctions. This option is used for junctions that were created with the type of `tcp` or `ssl`.

-P *port*

Specifies the TCP port of the proxy server. The default value is 7138.

For *port*, use any valid port number. A valid port number is any positive number that is allowed by TCP/IP and that is not currently being used by another application. Use the default port number value, or use a port number that is greater than 1000 that is not being used.

This option is used for junctions that were created with the type of `tcpproxy` or `sslproxy`.

-q *path*

Specifies the relative path for the **query_contents** script. By default, Security Verify Access looks for this script in the `/cgi_bin` subdirectory. If this directory is different or the **query_contents** file is renamed, use this option to indicate to WebSEAL the new URL to the file. Required for Windows virtual hosts.

This option is valid for all junction types except `localtcp` and `localssl`.

-u *uuid*

Specifies the UUID of this server when connected to WebSEAL over a stateful junction that was using the –s option. This option is used for junctions that were created with the type of `tcp` or `ssl`.

-w

Indicates Microsoft Windows 32 bit (Win32) file system support.

This option is used for junctions that were created with the type of `tcp` or `ssl`.

-y priority

The priority for the server (1-9). Default is 9.

vhost_label

Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name-instance_name*/@*vhost_label* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example adds a server with host name `xyz.ibm.com` to an existing virtual host junction with the label `support-vhost-http`, on the WebSEAL server `abc.ibm.com`:

```
pdadmin> server task default-webseald-abc.ibm.com virtualhost add \
-h xyz.ibm.com support-vhost-http
```

See also

“server task virtualhost create” on page 155
“server task virtualhost delete” on page 161
“server task virtualhost list” on page 162
“server task virtualhost remove” on page 167
“server task virtualhost show” on page 169

server task virtualhost create

Creates a virtual host junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

For local junctions:
server task *instance_name*-webseald-*host_name* `virtualhost create` –t *type* –d *dir* –v *virtual_host_name* [*options*] *vhost_label*

For non-local junctions:
server task *instance_name*-webseald-*host_name* `virtualhost create` –t *type* –h *host_name* [*options*] *vhost_label*

Options

–d *dir*
Specifies the local directory for a local virtual host junction.

This option is required for `localtcp` and `localssl` junction types.

–h *host_name*
Specifies the DNS host name or IP address of the target server. This option is valid only for non-local junctions; local junctions do not need a host name. Valid values for *host_name* include any valid IP host name. For example:

```
www.example.com
```

–t *type*
Specifies the type of virtual host junction. This option is required and must be one of the following types:

- `tcp`
- `tcpproxy`
- `ssl`
- `sslproxy`
- `localtcp`
- `localssl`

–v *virtual_host_name*[:*port*]

WebSEAL selects a virtual host junction to process a request if the HTTP **Host** header of the request matches:

- The virtual host name by the **-v** option and
- The port number that is specified by the **-v** option.

The –v option is also used to specify the value of the **Host** header of the request sent to the server.

The port number is required if the virtual host uses a non-standard port for the protocol. Standard port for TCP is 80; standard port for SSL is 443.

If –v is not specified for `tcp`, `ssl`, `tcpproxy`, and `sslproxy` type junctions, the junction is selected from the information that is contained in the –h *host_name* and –p *port* options.

The –v option is required for `localtcp` and `localssl` type junctions

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

options

Specifies the options that you can use with the **`server task virtualhost create`** command. (Optional) These options include:

–A

Enables a virtual host junction to support the lightweight third-party authentication mechanism (LTPA). This option requires the –F and –Z options. The –A, –F, and –Z options all must be used together.

This option is valid for all junction types except `localtcp` and `localssl`.

–2

You can use this option with the –A option to specify that LTPA version 2 cookies (LtpaToken2) are used. The –A option without the –2 option specifies that LTPA version 1 cookies (LtpaToken) are used.

–b *BA_value*

Defines how the WebSEAL server passes client identity information in HTTP basic authentication (BA) headers to the virtual host, which is one of the following values:

- `filter`
- `ignore`
- `supply`
- `gso`

This option is valid for all junction types except `localtcp` and `localssl`. The default value is `filter`.

–B

Indicates that WebSEAL uses the BA header information to authenticate to the virtual host and to provide mutual authentication over SSL. This option requires the `–U` and `–W` options.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–c *header_type*

Inserts the Security Verify Access client identity in HTTP headers across the virtual host junction. The *header_type* argument can include any combination of the listed Security Verify Access HTTP header types:

- {iv_user|iv_user-l}
- iv_groups
- iv_creds
- all

The header types must be comma-separated, and cannot have a space between the types. For example: `-c iv_user,iv_groups`

Specifying `–c all` is the same as specifying `–c iv_user,iv_groups,iv_creds`.

This option is valid for all junction types except `localtcp` and `localssl`.

–C

Supports mutual authentication by enabling the front-end WebSEAL server to pass its identity information to the back-end WebSEAL server. The front-end WebSEAL server passes information in a Basic Authentication (BA) header. Additionally, the –C option enables the single sign-on functionality that is provided by the –c option.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–D "*dn*"

Specifies the distinguished name of the server certificate. This value, matched with the actual certificate DN, enhances authentication and provides mutual authentication over SSL. For example, the certificate for `www.example.com` might have a DN of

```
"CN=WWW.EXAMPLE.COM,OU=Software,O=example.com\, Inc,L=Austin,
ST=Texas,C=US"
```

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

–e encoding_type

Specifies the encoding to use when HTTP headers is generated for virtual host junctions. This encoding applies to headers that are generated with both the –c junction option and tag-value. Possible values for encoding are as follows:

utf8_bin

WebSEAL sends the headers in UTF-8.

utf8_uri

WebSEAL sends the headers in UTF-8 but URI also encodes them. This behavior is the default behavior.

lcp_bin

WebSEAL sends the headers in the local code page of the WebSEAL server.

lcp_uri

WebSEAL sends the headers in the local code page of the WebSEAL server, but URI also encodes them.

This option is valid for all junction types except `localtcp` and `localssl`.

–f

Forces the replacement (overwrite) of an existing virtual host junction.

This option is used for junctions that were created with any junction type.

–F "*keyfile*"

Specifies the location of the key file that is used to encrypt LTPA cookie data.

The –F option requires –A and –Z options. The –A, –F, and –Z options all must be used together.

This option is valid for all junction types except `localtcp` and `localssl`.

-g *vhost_label*

The –g option causes a second more virtual host junction to share a protected object space as the initial virtual host junction.

This option is appropriate for junction pairs only (two junctions by using complementary protocols). The option does not support the association of more than two junctions.

-H *host_name*

Specifies the DNS host name or IP address of the proxy server. The –P option also supports proxy server junctions. Valid values for *host_name* include any valid IP host name. For example:

```
proxy.www.example.com
```

This option is valid only with junctions that were created with the type of `tcpproxy` or `sslproxy`.

-i

Indicates that the WebSEAL junction does not treat URLs as case-sensitive. To correctly authorize requests for junctions that are not case-sensitive, WebSEAL does the authorization check on a lowercase version of the URL. For example, a web server that is running on a Windows operating system treats requests for `INDEX.HTM` and `index.htm` as requests for the same file.

Junctions to such a web server must be created with the `–i` or `–w` option. ACLs or POPs that are attached to objects beneath the junction point must use the lowercase object name. An ACL attached to `/junction/index.htm` applies to all the following requests if the `–i` or `–w` option is used:

```
/junction/INDEX.HTM
/junction/index.htm
/junction/InDeX.HtM
```

This option is valid for all junction except for the type of `localtcp` and `localssl`. Local junctions are not case-sensitive only on Win32 platforms; all other platforms are case-sensitive.

-k

Sends WebSEAL session cookies to the back-end virtual host. By default, cookies are removed from requests that are sent to the server.

This option is valid for all junction types except `localtcp` and `localssl`.

-K "*key_label*"

Specifies the key label of the client-side certificate that WebSEAL must present to the server. Use of this option allows the virtual host to authenticate the WebSEAL server by using client certificates.

This option is valid only with junctions that were created with the type of `ssl` and `sslproxy`.

-l *percent*

Defines the soft limit for consumption of worker threads.

This option is valid for all junction types except `localtcp` and `localssl`.

-L *percent*

Defines the hard limit for consumption of worker threads.

This option is valid for all junction types except `localtcp` and `localssl`.

-p *port*

Specifies the TCP port of the third-party server. The default value is `80` for TCP junctions and 443 for SSL junctions.

This option is valid for all junction types except `localtcp` and `localssl`.

-P *port*

Specifies the TCP port number for the HTTP proxy server. The –P option is required when the –H option is used.

This option is valid only with junctions that were created with the type of `tcpproxy` or `sslproxy`.

-q -S

Specifies the relative path for the `query_contents` script. By default, Security Verify Access looks for the `query_contents` script in the `/cgi_bin` directory. If this directory is different or the `query_contents` file name is renamed, this option indicates to WebSEAL the new URL to the file. Required for Windows virtual hosts.

This option is valid for all junction types except `localtcp` and `localssl`.

-r

Inserts the incoming IP address into the HTTP header across the junction.

This option is valid for all junction types except `localtcp` and `localssl`.

-R

Allows the request to proceed but provides the rule failure reason to the junction in an HTTP header. If the –R option is not used and a rule failure occurs, WebSEAL does not allow the request to proceed.

This option is valid for all junction types except `localtcp` and `localssl`.

-s

Indicates that the virtual host junction support stateful applications. By default, virtual host junctions are not stateful.

This option is valid for all junction types except `localtcp` and `localssl`.

-S

Indicates the location of the forms single sign-on configuration file.

This option is valid for all junction types except `localtcp` and `localssl`.

-T {*resource* | *resource_group*}

Specifies the name of the GSO resource or resource group. This option is required only when the –b gso option is used.

This option is valid for all junction types except `localtcp` and `localssl`.

-u *uuid*

Specifies the Universally Unique Identifier (UUID) of a server that is connected to WebSEAL by using a stateful virtual host junction (–s option).

This option is valid for all junction types except `localtcp` and `localssl`.

-U "*user_name*"

Specifies the WebSEAL server user name. This option requires the –B and –W options. WebSEAL uses the BA header information to authenticate to the virtual host and to provide mutual authentication over SSL.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

-w

Indicates Microsoft Windows 32 bit (Win32) file system support. This option provides all the functionality that is provided by the `–i` junction option. The option disallows requests that contain file names that might be interpreted as Win32 file name aliases.

This option is valid for all junction types except `localtcp` and `localssl`. Local junctions prohibit URLs that contain Win32 file name aliases on Win32 but allow such URLs on other platforms.

-W "*password*"

Specifies the WebSEAL server password. This option requires the –B and –U options. WebSEAL uses the BA header information to authenticate to the virtual host and to provide mutual authentication over SSL.

This option is valid only with junctions that were created with the type of `ssl` or `sslproxy`.

-y priority

The priority for the server (1-9). Default is 9.

–Y

Enables the Federation Runtime single sign-on (SSO) for the junction.

Note: Before you use this option, you must first configure the WebSEAL configuration file to support the Federation Runtime single sign-on over junctions.

–z *replica_set*

Specifies the replica set, as follows:

For SMS environments:

Sessions on the virtual host junction are managed under the specified replica set. Used to group or separate login sessions among multiple virtual hosts.

For non-SMS environments:

Sessions on the virtual host junction are managed under the specified replica set. Controls the partitioning of the WebSEAL session cache. The virtual host can be part of the same replica set as any standard junction that is assigned to that same replica set. Standard junctions are assigned to replica sets through the `standard-junction-replica-set` entry of the `[session]` stanza.

–Z *keyfile_pwd*

Specifies the password of the key file that is used to encrypt LTPA cookie data. This option requires the –A and –F options. The –A, –F, and –Z options all must be used together.

This option is valid for all junction types except `localtcp` and `localssl`.

vhost_label

Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **s** (server administration) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/@`*`vhost_label`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

For more information about gathering statistics, see the Troubleshooting topics in the IBM Knowledge Center.

Example

The following example creates an SSL type virtual host junction with the `vhost-xy-https` label. This junction serves the virtual host `x.y.com` on the junctioned server `cruz1.ibm.com`. WebSEAL responds to the `Host: x.y.com` header in SSL (HTTPS) requests by forwarding the requests across this virtual host junction:

```
pdadmin> server task default-webseald-abc.ibm.com virtualhost create \
-t ssl -h cruz1.ibm.com -v x.y.com vhost-xy-https
```

See also

“server task virtualhost add” on page 153
“server task virtualhost delete” on page 161
“server task virtualhost list” on page 162
“server task virtualhost remove” on page 167
“server task virtualhost show” on page 169

server task virtualhost delete

Deletes a virtual host junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* `virtualhost delete` *`vhost_label`*

Description

The **`server task virtualhost delete`** command deletes a virtual host junction. A virtual host junction cannot be deleted if a second virtual host junction refers to it through the –g option. An error message is returned at such an attempt.

Options

`instance_name`*`-webseald-`*`host_name`
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named `web2`, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

`vhost_label`
Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/@`*`vhost_label`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example deletes the virtual host junction support-vhost-https from the WebSEAL server abc.ibm.com:

```
pdadmin> server task default-webseald-abc.ibm.com virtualhost delete \
support-vhost-https
```

See also

"server task virtualhost add" on page 153
"server task virtualhost create" on page 155
"server task virtualhost list" on page 162
"server task virtualhost remove" on page 167
"server task virtualhost show" on page 169

server task virtualhost list

Lists all configured virtual host junctions by label name.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost list

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

Authorization

Users and groups that require access to this command must be given the **l** (list) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/@*per_vhost_label* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
: The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

  See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example lists the label names of all virtual host junctions that are configured on the `abc.ibm.com` WebSEAL server:

```
pdadmin> server task default-webseald-abc.ibm.com virtualhost list
```

Output is like:

```
pubs-vhost-http
sales-vhost-https
support-vhost-http
```

See also

“server task virtualhost add” on page 153
“server task virtualhost create” on page 155
“server task virtualhost delete” on page 161
“server task virtualhost remove” on page 167
“server task virtualhost show” on page 169

server task virtualhost offline

Places the server that is at this virtual host junction in an offline operational state.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost offline [–i *server_uuid*] *vhost_label*

Description

The **server task virtualhost offline** command places the server that is at this virtual host junction in an offline operational state. No additional requests are sent to the specified server. If a server is not specified, all servers that are at this virtual host junction are placed in an offline operational state.

Options

–i *server_uuid*
: Specifies the UUID of the server to place in an offline operational state. If a server is not specified, all servers that are at this virtual host junction are placed in an offline operational state. Use the **server task virtualhost show** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name
: Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

  The *instance_name* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

vhost_label

Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`*`/@`*`vhost_label`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Examples

In the following example:

- The virtual host junction:
  - Has the label `support-vhost-https`.
  - Is configured on the WebSEAL server `abc.ibm.com`.
  - Supports the virtual host `support.ibm.com`.
- The virtual host `support.ibm.com` is on the junctioned server `int3.ibm.com`.

There is a requirement to place the `int3.ibm.com` server in an offline operational state. To determine the UUID of this junctioned server, run the **`server task virtualhost show`** command:

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost show support-vhost-https
```

Output is like:

```
Virtual Host label: support-vhost-https
Type: SSL
...
Virtual hostname: support.ibm.com
Alias: ibm.com
Alias: support
Virtual Host junction protocol partner: support-vhost-http
Server 1:
ID: bacecc66-13ce-11d8-8f0a-09267ea5aa77
Server State: running
Operational State: Throttled
Throttled at: 2005-03-01-17:07:24
```

```
Hostname: int3.ibm.com
Port: 443
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Current requests: 0
Total requests: 1
```

Place this server in an offline operational state by using the following command:

```
pdadmin> server task default-webseald-cruz virtualhost offline \
-i bacecc66-13ce-11d8-8f0a-09267ea5aa77 support-vhost-https
```

See also

“server task offline” on page 123
“server task online” on page 125
“server task throttle” on page 149
“server task virtualhost online” on page 165
“server task virtualhost throttle” on page 170

server task virtualhost online

Places the server that is at this virtual host junction in an online operational state.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost online [–i *server_uuid*] *vhost_label*

Description

The **server task virtualhost online** command places the server that is at this virtual host junction in an online operational state. The server now resumes normal operation. If a server is not specified, all servers that are at this virtual host junction are placed in an online operational state.

Options

-i *server_uuid*
UUID of the server to place in an online operational state. If a server is not specified, all servers that are at this virtual host junction are placed in an online operational state. Use the **server task virtualhost show** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

vhost_label
Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name-instance_name*/@*vhost_label* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

In the following example:

- The virtual host junction:
  - Has the label support-vhost-https.
  - Is configured on the WebSEAL server abc.ibm.com.
  - Supports the virtual host support.ibm.com
- The virtual host support.ibm.com is on the junctioned server int3.ibm.com.

There is a requirement to place the int3.ibm.com server in an online operational state. To determine the UUID of this junctioned server, run the **server task virtualhost show** command:

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost show support-vhost-https
```

Output is like:

```
Virtual Host label: support-vhost-https
Type: SSL
...
Virtual hostname: support.ibm.com
Alias: ibm.com
Alias: support
Virtual Host junction protocol partner: support-vhost-http
Server 1:
ID: bacecc66-13ce-11d8-8f0a-09267ea5aa77
Server State: running
Operational State: Offline
Hostname: int3.ibm.com
Port: 443
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Current requests: 0
Total requests: 1
```

Place this server in an online operational state by using the following command:

```
pdadmin> server task default-webseald-cruz virtualhost online \
-i bacecc66-13ce-11d8-8f0a-09267ea5aa77 support-vhost-https
```

See also

“server task offline” on page 123
“server task online” on page 125
“server task throttle” on page 149
“server task virtualhost offline” on page 163
“server task virtualhost throttle” on page 170

server task virtualhost remove

Removes the specified server from a virtual host junction.

Requires authentication (administrator ID and password) to use this command.

Syntax

`server task` *`instance_name`*`-webseald-`*`host_name`* `virtualhost remove –i` *`server_uuid`* *`vhost_label`*

Options

`–i` ***`server_uuid`***
Specifies the UUID of the server to be removed from the virtual host junction. For this command, the `–i` option, normally used to treat URLs as case-sensitive, operates like the `–u` option. See the **`server task show`** command for details about obtaining the UUID.

`instance_name`**`-webseald-`*****`host_name`***
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **`server list`** command.

The *`instance_name`* specifies the configured name of the WebSEAL server instance. The `webseald` designation indicates that the WebSEAL service performs the command task. The *`host_name`* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is `default`. The host computer name where the WebSEAL server is installed is `abc.ibm.com`. Then, the full WebSEAL server name is `default-webseald-abc.ibm.com`.

If an additional WebSEAL server instance is configured and named `web2`, the full WebSEAL server name is `web2-webseald-abc.ibm.com`.

`vhost_label`
Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the `/WebSEAL/`*`host_name`*`-`*`instance_name`*`/@`*`vhost_label`* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
: The command failed. See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The following example removes the junctioned server `int4.ibm.com` from the virtual host junction `support-vhost-https`. To determine the UUID of the server to be removed, run the **`server task virtualhost show`** command:

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost show support-vhost-https
```

Output is like:

```
Virtual Host label: support-vhost-https
Type: SSL
Junction hard limit: 0 - using global value
Junction soft limit: 0 - using global value
Active worker threads: 0
Basic authentication mode: filter
Forms based SSO: disabled
Authentication HTTP header: do not insert
Remote Address HTTP header: do not insert
Stateful junction: no
Boolean Rule Header: no
Delegation support: no
Mutually authenticated: no
Insert WebSphere LTPA cookies: no
Insert WebSEAL session cookies: no
Request Encoding: UTF-8, URI Encoded
Virtual hostname: support.ibm.com
Alias: ibm.com
Alias: support
Virtual Host junction protocol partner: support-vhost-http
Server 1:
ID: bacecc66-13ce-11d8-8f0a-09267ea5aa77
Server State: running
Hostname: int3.ibm.com
Port: 443
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Total requests: 1
Server 2:
ID: xycecc77-19ve-81y5-4h0a-90267hj5nn57
Server State: running
Hostname: int4.ibm.com
Port: 444
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Total requests: 1
```

Remove the server from the virtual host junction:

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost remove -i xycecc77-19ve-81y5-4h0a-90267hj5nn57 support-vhost-https
```

See also

“server task virtualhost add” on page 153
“server task virtualhost create” on page 155
“server task virtualhost delete” on page 161
“server task virtualhost list” on page 162
“server task virtualhost show” on page 169

server task virtualhost show

Displays information about the specified virtual host junction. The virtual host junction must exist, or an error is displayed.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost show *vhost_label*

Options

instance_name*-webseald-*host_name

Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

vhost_label

Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **l** (list) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/@*vhost_label* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

The example shows information for the virtual host junction:

- With the label support-vhost-https.
- Configured on the WebSEAL server abc.ibm.com.
- That supports the virtual host support.ibm.com.

The virtual host `support.ibm.com` is on the junctioned server `int3.ibm.com`.

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost show support-vhost-https
```

Output is like:

```
Virtual Host label: support-vhost-https
Type: SSL
Junction hard limit: 0 - using global value
Junction soft limit: 0 - using global value
Active worker threads: 0
Basic authentication mode: filter
Forms based SSO: disabled
Authentication HTTP header: do not insert
Remote Address HTTP header: do not insert
Stateful junction: no
Boolean Rule Header: no
Delegation support: no
Mutually authenticated: no
Insert WebSphere LTPA cookies: no
Insert WebSEAL session cookies: no
Request Encoding: UTF-8, URI Encoded
Virtual hostname: support.ibm.com
Alias: ibm.com
Alias: support
Virtual Host junction protocol partner: support-vhost-http
Server 1:
ID: bacecc66-13ce-11d8-8f0a-09267ea5aa77
Server State: running
Hostname: int3.ibm.com
Port: 443
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Total requests: 1
```

See also

"server task virtualhost add" on page 153
"server task virtualhost create" on page 155
"server task virtualhost delete" on page 161
"server task virtualhost list" on page 162
"server task virtualhost remove" on page 167

server task virtualhost throttle

Places the server that is at this virtual host junction in a throttled operational state.

Requires authentication (administrator ID and password) to use this command.

Syntax

server task *instance_name*-webseald-*host_name* virtualhost throttle [–i *server_uuid*] *vhost_label*

Description

The **server task virtualhost throttle** command places the server that is at this virtual host junction in a throttled operational state. Users can create a session with WebSEAL before the invocation of this command. Only requests from such users continue to be processed by the specified server. If a server is not specified, all servers that are at this virtual host junction are placed in a throttled operational state.

Options

-i *server_uuid*
Specifies the UUID of the server to throttle. If a server is not specified, all servers that are at this virtual host junction are placed in a throttled operational state. Use the **server task virtualhost show** command to determine the ID of a specific server. (Optional)

instance_name*-webseald-*host_name
Specifies the full server name of the installed WebSEAL server instance. You must specify this full server name in the exact format as displayed in the output of the **server list** command.

The *instance_name* specifies the configured name of the WebSEAL server instance. The webseald designation indicates that the WebSEAL service performs the command task. The *host_name* is the name of the physical computer where the WebSEAL server is installed.

For example, the configured name of a single WebSEAL server instance is default. The host computer name where the WebSEAL server is installed is abc.ibm.com. Then, the full WebSEAL server name is default-webseald-abc.ibm.com.

If an additional WebSEAL server instance is configured and named web2, the full WebSEAL server name is web2-webseald-abc.ibm.com.

vhost_label
Specifies the label name of the virtual host junction.

Authorization

Users and groups that require access to this command must be given the **c** (control) permission in the ACL that governs the /WebSEAL/*host_name*-*instance_name*/@*vhost_label* object. For example, the **sec_master** administrative user is given this permission by default.

Return codes

0
The command completed successfully.

Note: For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not be able to successfully complete the command. The WebSEAL server returns an error message.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: This command is available only when WebSEAL is installed.

Example

In the following example, the virtual host junction:

- Has the label support-vhost-https.
- Is configured on the WebSEAL server abc.ibm.com.
- Supports the virtual host support.ibm.com.

The virtual host support.ibm.com is on the junctioned server int3.ibm.com.

There is a requirement to place the `int3.ibm.com` server in a throttled operational state. To determine the UUID of this junctioned server, run the **`server task virtualhost show`** command:

```
pdadmin> server task default-webseald-abc.ibm.com \
virtualhost show support-vhost-https
```

Output is like:

```
Virtual Host label: support-vhost-https
Type: SSL
...
Virtual hostname: support.ibm.com
Alias: ibm.com
Alias: support
Virtual Host junction protocol partner: support-vhost-http
Server 1:
ID: bacecc66-13ce-11d8-8f0a-09267ea5aa77
Server State: running
Operational State: Online
Hostname: int3.ibm.com
Port: 443
Server DN:
Query_contents URL: /cgi-bin/query_contents
Query-contents: unknown
Case insensitive URLs: no
Allow Windows-style URLs: yes
Current requests: 0
Total requests: 1
```

Place this server in a throttled operational state by using the following command:

```
pdadmin> server task default-webseald-cruz virtualhost throttle \
-i bacecc66-13ce-11d8-8f0a-09267ea5aa77 support-vhost-https
```

See also

“server task throttle” on page 149
“server task offline” on page 123
“server task online” on page 125
“server task virtualhost offline” on page 163
“server task virtualhost online” on page 165

server task server restart

Restarts a WebSEAL server by using the Security Verify Access server task framework.

This command requires authentication of administrator ID and password.

Syntax

`server task` *`server_name`* `server restart`

Options

`server_name`
Specifies the name of the Security Verify Access authorization server restart.

Authorization

Users and groups that require access to this command must have the **s** (administration) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`* object. For example, the **sec_master** administrative user has this permission by default.

Return codes

0
: The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not successfully complete the command. The WebSEAL server returns an error message.

1
: The command failed. When a command fails, the **`pdadmin`** command provides a description of the error and an error status code in hexadecimal format, for example, `0x14c012f2`. See "Error messages" in the IBM Knowledge Center for a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: The restart is successful only if the WebSEAL server was started by using the **pdweb_start** script. The script must be running as a daemon on AIX, Linux, or Solaris, or as a service on Windows). The restart command does not work if the WebSEAL server is running in the foreground.

The result of the restart is not displayed on the administration console. You must examine the WebSEAL log files to confirm that the server restart was successful.

Example

The following example restarts `server03`:

```
pdadmin> server task server03 server restart
```

server task server sync

Synchronizes configuration data between two WebSEAL servers by using the Security Verify Access server task framework.

This command requires authentication of administrator ID and password.

Syntax

`server task` *`webseal_server`* `server sync` *`server_name`*

Options

`webseal_server`
: Specifies the fully qualified server name of the installed WebSEAL instance.

`server_name`
: Specifies the name of the Security Verify Access authorization server from which data is extracted. Configuration data on the host system is backed up and then synchronized with this data.

Authorization

Users and groups that require access to this command must have the **s** (administration) permission in the ACL that governs the `/WebSEAL/`*`host_name-instance_name`* object. For example, the **sec_master** administrative user has this permission by default.

Return codes

0
: The command completed successfully. For WebSEAL **`server task`** commands, the return code is `0` when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format, for example, 0x14c012f2. See "Error messages" in the IBM Knowledge Center for a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example synchronizes configuration data with server default-webseald-abc.ibm.com:

```
pdadmin> server task default-webseald-abc.ibm.com server sync
master-webseald-abc.ibm.com
```

server task file cat

Returns the contents of a specified file to the administration console. This command requires authentication of administrator ID and password.

Syntax

server task *server_name* file cat *file_name* *byte_offset* [-max *bytes*] [-encode]

Options

server_name

Specifies the name of the Security Verify Access authorization server on which the file is hosted.

file_name

Specifies the fully qualified name of the file which is to be retrieved.

byte_offset

Specifies the offset in bytes from the start of the file at which the data is retrieved.

-max *bytes*

Specifies the maximum number of bytes to be returned from the file. If you do not specify this parameter, the maximum number of bytes returned is controlled by the max-file-cat-command-length value in the [server] stanza. The max-file-cat-command-length value takes precedence over the -max *bytes* value. If the specified file is larger than the max-file-cat-command-length value, the returned data is truncated.

--encode

Designates that the contents of the file must be base-64 encoded before it is returned. (Optional)

Authorization

Users and groups that require access to this command must have the **s** (administration) permission in the ACL that governs the /WebSEAL/*host_name-instance_name* object. For example, the **sec_master** administrative user has this permission by default.

Return codes

0

The command completed successfully. For WebSEAL **server task** commands, the return code is 0 when the command is sent to the WebSEAL server without errors. However, even after the command was successfully sent, the WebSEAL server might not successfully complete the command. The WebSEAL server returns an error message.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format, for example, 0x14c012f2. See "Error messages" in the IBM Knowledge Center for a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Note: For more information about WebSEAL junctions, see the Administering topics in the IBM Knowledge Center.

This command is available only when WebSEAL is installed.

Example

The following example requests the first 512 bytes of data (base-64 encoded) from the `readme.txt` file. The file is in the `/temp/` folder on `server03`:

```
pdadmin> server task server03 file cat /temp/readme.txt 0 -max 512 -encode
```

The output is the content of the specified file.

user create

Creates a Security Verify Access user.

Requires authentication (administrator ID and password) to use this command.

Syntax

`user create` [`–gsouser`] [`–no-password-policy`] [`–account-valid`] *`user_name dn cn sn password`* [*`groups`*]

Description

A *user* is a registered participant of the secure domain. A *GSO user* is a Security Verify Access user that also has the authority to use single sign-on to work with web resources.

You can create users in the Active Directory Lightweight Directory Service (AD LDS) user registry. Create such users in the same AD LDS partition where the Security Verify Access Management Domain information is stored.

The `–gsouser` option enables global sign-on capabilities. Users that are created in an Active Directory are automatically given the capability to own single sign-on credentials. This capability cannot be removed. When you use an LDAP user registry, this capability must be explicitly granted. After this capability is granted, it can be removed.

The `–no-password-policy` option allows the administrator to create the user with an initial password that is not checked by the existing global password policies. If this option is not present in the command, the password that is provided is checked against the global password policies. In this case, the **`user create`** command fails if the password is invalid, and the error message includes information about what conditions were not met.

However, if the administrator applies the `password` option on the **`user modify`** command, the `-no-password-policy` option is not available. Therefore, the modified password is always checked against the global password policy settings.

The `–account-valid` option allows the administrator to mark the account as valid. If this option is not present in the command, the account is left in the invalid state. To make the user account valid, you must use the **`user modify`** command to set the `account-valid` option to `yes`.

Options

`-gsouser`
: Enables the global sign-on (GSO) capabilities for the user. Applies only to users created in an LDAP user registry.

`-no-password-policy`
: Indicates that the password policy is not enforced during the creation of the user account. The nonenforcement does not affect password policy enforcement after user creation. (Optional)

-account-valid
Indicates that the account is valid. (Optional)

cn
Specifies the common name that is assigned to the user that is being created. For example: "Mary"

dn
Specifies the registry identifier that is assigned to the user that is being created. The registry identifier must be known before a new user account can be created. The registry identifier must be unique within the user registry. If the user registry is Active Directory, certain characters are not allowed. See "Characters disallowed for distinguished names" on page 187 for the list of these characters.

The format for a distinguished name is like:

```
"cn=Mary Jones,ou=Austin,o=Tivoli,c=us"
```

groups
Specifies a list of groups to which the new user is assigned. The format of the group list is a parenthesized list of group names, which are separated by spaces. The groups must exist, or an error is displayed. Examples of groups: deptD4D and printerusers. (Optional)

password
Specifies the password that is set for the new user. Passwords must adhere to the password policies set by the administrator.

sn
Specifies the short name of the user that is being created. For example: "Jones"

user_name
Specifies the name for the user to create. This name must be unique. A valid username is an alphanumeric string that is not case-sensitive. If the user registry is Active Directory, certain characters are not allowed. See "Characters disallowed for user and group name" on page 186 for the list of these characters. If the user is a GSO user, certain characters are not allowed. See "Characters disallowed for GSO names" on page 188 for the list of these characters.

Note: Consider that you did not change the 7 - bit checking default value during configuration of the Sun web server. In this case, turn off checking so that non-ASCII characters can be stored in attributes.

Examples of usernames are dlucas, sec_master, "Mary Jones".

Return codes

0
The command was completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Documentation. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example, entered as one line, creates user dlucas:

```
pdadmin sec_master> user create –gsouser dlucas "cn=Diana
Lucas,ou=Austin,o=Tivoli,c=US" "Diana Lucas" Lucas lucaspwd
```

- The following example, entered as one line, creates user maryj:

```
pdadmin sec_master> user create –gsouser maryj "cn=Mary Jones,o=tivoli,c=us"
Mary Jones maryjpw
```

See also

“user delete” on page 177
“user import” on page 177
“user modify” on page 180

user delete

Deletes the specified Security Verify Access user. Optionally deletes the information of the user in the user registry.

Requires authentication (administrator ID and password) to use this command.

Syntax

`user delete` [`-registry`] *`user_name`*

Options

`-registry`
Deletes the information of the user from the user registry. If this option is not specified, the registry user information can be used to create another Security Verify Access user by using the **`user import`** command. (Optional)

`user_name`
Specifies the name of the account to be deleted. Any resource credentials that are associated with a user account are automatically removed at the same time the user account is deleted. The user must exist, or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0
The command completed successfully.

1
The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example deletes the `dlucas` user:

```
pdadmin sec_master> user delete dlucas
```

See also

“user create” on page 175
“user import” on page 177

user import

Creates a Security Verify Access user by importing user data that exists in the user registry.

Requires authentication (administrator ID and password) to use this command.

If the user registry is Active Directory Lightweight Directory Service (AD LDS), import within the AD LDS partition where the Security Verify Access management domain information is stored.

Syntax

user import [–gsouser] *user_name dn* [*group_name*]

Description

Imported user accounts are created invalid by default. To make the user account valid, you must use the **user modify** command to set the `account-valid` option to `yes`.

Options

-gsouser

Specifies that the user has single sign-on capabilities. (Optional)

dn

Specifies the registry identifier of the user that is being imported. This identifier must exist in the user registry and must not be associated with another user in the same Security Verify Access secure domain. The format for a distinguished name is like:

```
cn=Claude Wright,ou=Austin,o=Tivoli,c=us
```

group_name

Specifies an optional group to which the user is being added. The group must exist, or an error is displayed. (Optional)

Examples of group names are `Credit`, `Sales`, and `Test-group`.

user_name

Specifies a unique Security Verify Access user name. This user is created from information that exists in the user registry. A valid user name is an alphanumeric string that is not case-sensitive. If the user is a GSO user, certain characters are not allowed. See "Characters disallowed for GSO names" on page 188 for the list of these characters.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Example

The following example creates the user `mlucaser` by importing information from the registry user `cn=Mike Lucaser,ou=Austin,o=Tivoli,c=US`:

```
pdadmin sec_master> user import –gsouser mlucaser
"cn=Mike Lucaser,ou=Austin,o=Tivoli,c=US"
```

See also

"user create" on page 175
"user modify" on page 180

user list

Lists users by Security Verify Access user name or by registry identifier.

Requires authentication (administrator ID and password) to use this command.

Syntax

user list *pattern max_return*

user list-dn *pattern max_return*

Options

list *pattern max_return*

Specifies the pattern for the principal name. The pattern can include a mixture of wildcard and string constants. The specified pattern is case-sensitive. For example: `*luca*`

The *pattern max_return* options specify the maximum number of entries that are found and returned for a single request. The number that is returned is also governed by the server configuration, which specifies the maximum number of results that can be returned as part of a search operation.

The actual maximum returned entries is the minimum number of results between the *pattern max_return* and the configured value on the server. The configured value is taken from the `max-search-size=[0|`*num_entries*`]` entry in the `[ldap]` stanza. The `[ldap]` stanza is in the `ldap.conf` configuration file.

list-dn *pattern max_return*

Specifies the pattern for the common name (CN) portion of the registry identifier of the user. When you specify the pattern, you can exclude the `cn=` component. The pattern can include a mixture of wildcard and string constants, and is case-sensitive. For example, `*luca*`.

The returned list contains users that are defined in the user registry but are not necessarily Security Verify Access users. Users that are not Security Verify Access users can be imported into Security Verify Access by using the **user import** command.

Note: When the user registry contains many user definitions, use wildcard characters with discretion. When a pattern includes one or more wildcard characters, the command attempts to find all user definitions that match the specified pattern. However, the command displays only the specified number of matching definitions in the user registry.

For example, if the user registry contains 10,000 definitions, specifying a single wildcard (`user list * 100`) displays only the first 100 matching definitions but finds all 10,000 definitions.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`).

See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example lists the users that match the specified pattern:

  ```
  pdadmin sec_master> user list *luca* 2
  ```

  The output is like:

```
dlucas
mlucaser
```

- The following example lists the users that match the specified registry identifier:

  ```
  pdadmin sec_master> user list-dn *luca* 2
  ```

  The output is like:

  ```
  cn=Diana Lucas,ou=Austin,o=Tivoli,c=US
  cn=Mike Lucaser,ou=Austin,o=Tivoli,c=US
  ```

See also

"user show" on page 182

user modify

Changes various user account attributes.

Requires authentication (administrator ID and password) to use this command.

Syntax

user modify *user_name* account-valid {yes|no}

user modify *user_name* password *password*

user modify *user_name* password-valid {yes|no}

user modify *user_name* description *description*

user modify *user_name* gsouser {yes|no}

Options

account-valid {yes|no}
Enables or disables the specified user account. A user cannot log in with a disabled account. Valid values are yes and no.

password *password*
Modifies the user password. The new password must comply with password policies in effect.

When a password is set or changed, the password must comply to:

- The defined Security Verify Access password policy and
- The password policies for any underlying operating systems or user registry.
-

When the password policy is enforced, Security Verify Access first validates compliance against the Security Verify Access password policy currently in effect. Then, Security Verify Access validates compliance against the underlying user registry. Although a password complies to the defined Security Verify Access policy, it might fail against the password policy of the underlying user registry.

Note: Old passwords can still be used after a password change when:

- You are using Active Directory as your user registry.
- The Active Directory server is running on Windows 2003 SP1 or later.

For more information, see the following web page:

http://support.microsoft.com/?id=906305

password-valid {yes|no}

Validates or invalidates the password for the specified user account. Valid values are yes and no. If the value is no, the password seems expired and the user cannot log in using the password. For a user to log in, an administrator must set the valid state to yes. The user can also authenticate by using another method, such as using a certificate.

Another reason a user might not be able to authenticate with a specified password is because the maximum password age was exceeded. If you check and find that the password-valid is set to yes, then try changing the value for the **policy set max-password-age** parameter. Only an administrator or a user that has the authority can set the max-password-age policy on a user account. A user cannot set this policy on their own account. This policy sets the maximum time, in days, that a password is valid. Time is relative to the last time the password was changed.

When you change the value for password-valid or reset **policy set max-password-age**, you do not have to change the password.

If you reset a password, the password-valid parameter automatically switches to back to yes, and the max-password-age parameter resets the age to expire. For example, if the maximum password age is set to 30 days, another 30 days begins from the time you reset the password.

user_name

Specifies the name of the account to be modified. The user must exist, or an error is displayed. A valid user name is an alphanumeric string that is not case-sensitive. If the user is a GSO user, certain characters are not allowed. See "Characters disallowed for GSO names" on page 188 for the list of these characters. Examples of user names are dlucas, sec_master, and "Mary Jones"

description *description*

Specifies any text string that describes the user that is being created. Examples of user description are "Head of department" and "Department number of employee".

gsouser {yes|no}

Enables global sign-on (GSO) capabilities for the specified user. Valid values are yes and no.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, 0x14c012f2). See "Error messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example enables the specified user account:

```
pdadmin sec_master> user modify dlucas account-valid yes
```

- The following example changes the password for a user account:

```
pdadmin sec_master> user modify dlucas password newpasswd
```

See also

"user create" on page 175
"user import" on page 177

user show

Displays the properties of the specified user.

This command requires authentication of administrator ID and password.

Syntax

user show *user_name*

user show-dn *dn*

user show-groups *user_name*

Options

show user_name

Specifies the name of the user to display. The user must exist, or an error is displayed.

Based on the Policy Server and WebSEAL configuration settings, the following information is displayed:

```
Last login: YYYY-mm-dd-HH:MM:SS
Last Password Change: YYYY-mm-dd-HH:MM:SS
```

The system displays the local time of the computer where `pdadmin` was run. For more information about the last login and last password change configuration settings, see the Stanza Reference topics in the IBM Knowledge Center.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

show-dn dn

Displays the user that is specified by the identifier of the user in the user registry. The returned user is defined in the user registry, but it is not necessarily a Security Verify Access user. Users that are not Security Verify Access users can be imported into Security Verify Access by use of the **user import** command. The format for a distinguished name is like:

```
cn=Claude Wright,ou=Austin,o=Tivoli,c=us
```

Based on the Policy Server and WebSEAL configuration settings, the following information is displayed:

```
Last login: YYYY-mm-dd-HH:MM:SS
Last Password Change: YYYY-mm-dd-HH:MM:SS
```

The system displays the local time of the computer where `pdadmin` was run. For more information about last login and last password change configuration settings, see the Stanza Reference topics in the IBM Knowledge Center.

show-groups user_name

Displays the groups in which the specified user is a member. The user must exist, or an error is displayed.

Examples of user names are `dlucas`, `sec_master`, and `"Mary Jones"`.

Return codes

0

The command completed successfully.

1

The command failed. When a command fails, the **pdadmin** command provides a description of the error and an error status code in hexadecimal format (for example, `0x14c012f2`). See "Error

messages" in the IBM Knowledge Center. This reference provides a list of the Security Verify Access error messages by decimal or hexadecimal codes.

Examples

- The following example displays the user account information for `testuser`:

  ```
  pdadmin sec_master> user show testuser
  ```

  The output is like:

  ```
  Login ID: testuser
  LDAP DN: cn=testuser,o=tivoli,c=us
  LDAP CN: test
  LDAP SN: test
  Description: a test user
  Is SecUser: yes
  Is GSO user: no
  Account valid: no
  Password valid: yes
  Last login: 1999-09-05-01:08:55
  Last Password Change: 1999-09-04-05:06:45
  ```

- The following example displays the groups of which the specified user is a member:

  ```
  pdadmin sec_master> user show-groups dlucas
  ```

  The output is like:

  ```
  sales
  credit
  engineering
  ```

- The following example provides more information about the user when the registry identifier is specified:

  ```
  pdadmin sec_master> user show-dn "cn=Diana Lucas,ou=Austin,o=Tivoli,c=US"
  ```

  The output is like:

  ```
  Login ID: dlucas
  LDAP dn: cn=Diana Lucas,ou=Austin,o=Tivoli
  Inc,c=US
  LDAP cn: Diana Lucas
  LDAP sn: Lucas
  Description: Diana Lucas, Credit Dept HCUS
  IS SecUser: true
  IS GSO user: false
  Account valid: true
  Password valid: true
  Last login: 1999-09-05-01:08:55
  Last Password Change: 1999-09-04-05:06:45
  Authentication mechanism: Default:LDAP
  ```

See also

"user list" on page 179

Chapter 2. Password limitations and characters allowed in object names

When you specify Security Verify Access user names, group names, distinguished names, POP names, ACL policies, authorization rules, and domain names, certain characters might be disallowed.

Some factors that affect which characters are allowed are restrictions of the underlying user registry, server, or operating system.

This section describes the following limitations:

General password policies

You can change global user settings, such as password policies, login-failure policies, access policies, and account expiration policies. Additionally, you can override global password policies by setting individual password policies for the specified user.

For example, you can change a password policy so that the password policy:

- Is set only for a specific user.
- Overrides any password policy that is set globally for all users.

Using the Web Portal Manager or **pdadmin** commands, you can provide the following types of global password policies for all users:

- Minimum length that is allowed for a password
- Maximum age that is allowed for a password
- Minimum number of alphanumeric characters that are allowed in a password
- Minimum number of non-alphanumeric characters that are allowed in a password
- Maximum number of repeated characters that are allowed in a password
- Whether spaces are allowed in the password

By default, passwords must meet the following criteria:

- A minimum of eight alphanumeric characters, with a minimum of one number and four letters.
- A maximum of two repeated characters.

The valid range for minimum and maximum numbers can be any number. However, a reasonable number must be used for the task you are wanting to complete. For example, a minimum password length must:

- Be long enough to protect your system.
- Not be so short as to make it easy for someone to determine your password by trying different combinations.

Character limitations for passwords and user names

There are password characters that are valid, but must be treated differently when you run the **pdadmin** utility. These special characters have special meaning to the utility.

Enclose the password or user name in double quotation marks (") to escape the special character when:

- Setting or changing user passwords by using **user modify**.
- Logging in using **login**.

Otherwise, you receive an error message.

To escape the double quotation mark special character, enclose the password or user name in double quotation marks and use the backward slash (\) escape character. For example, to escape the password `abc"123`, type the string `"abc\"123"` in the **`pdadmin`** command when you type the password by using the −p option. When the interactive **`login`** command is used, no double quotation marks and escape character are needed.

The following special characters either must not be used or they must be escaped when using the **`pdadmin`** command:

- Comma (,)
- Double quotation (")
- Left parenthesis (()
- Number sign (#)
- Right parenthesis ())

Avoid the use of these characters as the first character in the password when setting or modifying the password with the **`user modify`** command:

- Hyphen (-)
- Left brace ({)
- Number sign (#)

Characters allowed for secure domain names

A valid local domain name is an alphanumeric, case-sensitive string. String characters are expected to be characters that are part of the local code set.

The following characters, numbers, and special characters can be used for secure domain names during use of the Web Portal Manager or **`pdadmin`** commands.

For example, for US English, secure domain names can contain a combination of the following characters:

- Letters (a-z A-Z)
- Numbers (0–9)
- Ampersand (&)
- Asterisk (*)
- At sign (@)
- Hyphen (-)
- Period (.)
- Plus sign (+)
- Underscore (_)

You cannot use a space in the domain name. The minimum and maximum lengths of the domain name, if there are limits, are imposed by the underlying registry.

Characters disallowed for user and group name

Environment aspects such as registries and command shells can affect special character handling. Because of the variability of special character handling in general, avoid the use of special characters.

Avoid the following character in user and group names that are defined by using distinguished name strings:

- Forward slash (/)

If Microsoft Active Directory is the user registry, care must be taken with user names and group names that contain the following character:

- Period (.)

A period (.) cannot be the last character of a user or group short name; for example: `jdoe.` and `jdoe.@my_ad_domain.com` are invalid user names.

If Microsoft Active Directory is the user registry, user names and group names can contain all Unicode characters except for the following characters:

- Asterisk (*)
- At sign (@)
- Colon (:)
- Equal sign (=)
- Forward slash (/)
- Left square bracket ([)
- Question mark (?)
- Right square bracket (])
- Vertical bar (|)
- Backward slash (\)
- Double quotation (")
- Left angle bracket (<)
- Right angle bracket (>)
- Plus sign (+)
- Semicolon (;)

Note: An at sign (@) is not allowed unless it is used to specify the domain. For example, `user@mydomain.com` is allowed; `user@name@mydomain.com` is not allowed.

The following characters are accepted in LDAP:

- Comma (,)
- Plus sign (+)
- Double quotation (")

  Note: Add a prefix with a backward slash (\) to escape any double quotation character in the user name.
- Backward slash (\)
- Left angle bracket (<)
- Right angle bracket (>)
- Semicolon (;)

If you use special characters with the **pdadmin** utility, enclose each argument of the user or group command with double quotation marks. The double quotation marks allow the argument to be entered without being subject to interpretation by the operating system shell command processor.

Because of the variability of special character handling in general, avoid the use of special characters.

Characters disallowed for distinguished names

Certain characters are treated differently by the different user registries.

In general, you can use special characters within a distinguished name (DN). However, certain special characters require an additional escape character. The following special characters must be escaped when used in a distinguished name:

- Comma (,)
- Plus sign (+)
- Semicolon (;)

Because of differences in registries and command shell processors, avoid the backward slash (\) character in distinguished names.

Characters disallowed for Microsoft Active Directory distinguished names

If Microsoft Active Directory is the user registry, certain special characters are not allowed in a distinguished name (DN). However, if the character is preceded by an additional escape character or is encoded in hexadecimal, then, it is allowed in a DN.

To encode in hexadecimal, replace the character with a backward slash (\) followed by two hexadecimal digits.

The following characters must be escaped by using the backward slash (\) character before they are used in a distinguished name:

- Number sign (#) at the beginning of the string
- A space at the end of the string
- Comma (,)
- Double quotation (")
- Left angle bracket (<)
- Plus sign (+)
- Right angle bracket (>)
- Semicolon (;)

Because of differences in registries and command shell processors, avoid the backward slash (\) character in distinguished names.

For other reserved characters, such as an equal sign (=), asterisk (*), or a non-UTF-8 character, the character must be encoded in hexadecimal.

Example 1

To create a user with a distinguished name that contains a comma next to the separator:

```
pdadmin sec_master> user create "johndoe"
"cn=doe\,john,cn=users,dc=mydomain,dc=com" John Doe password1
```

Example 2

To create a user with a distinguished name that contains a carriage return, which is a reserved character:

```
pdadmin sec_master> user create "johndoe"
"cn=doe\0DJohn,cn=users,dc=mydomain,dc=com" John Doe password1
```

The hexadecimal representation of a carriage return is `0D`.

Example 3

To create a user with a distinguished name that contains a number sign (#):

```
pdadmin sec_master>user create "#pounduser"
"cn=\#pounduser,cn=users,dc=mydomain,dc=com" "#pound" "user"
password1
```

Characters disallowed for GSO names

Certain characters are disallowed for GSO names.

You cannot use the following characters to create a global sign-on (GSO) user name, GSO resource name, or GSO resource group name:

- Asterisk (*)
- At sign (@)

- Backward slash (\)
- Colon (:)
- Comma (,)
- Double quotation (")
- Equal sign (=)
- Exclamation point (!)
- Left angle bracket (<)
- Left parenthesis (()
- Plus sign (+)
- Number sign (#)
- Right angle bracket (>)
- Right parenthesis ())
- Semicolon (;)
- Vertical bar (|)

It is possible to use most of these characters for other LDAP-related data. However, these characters have special meaning in LDAP DN syntax and filters. Examples of other LDAP-related data are: common name (CN), distinguished name (DN), and short name (SN) of a user.

Before you use any of these characters in user and group names, consult the documentation for your user registry to determine the effect of special characters.

Characters disallowed for authorization rule names

Certain characters are disallowed for authorization rule names.

These characters cannot be used in the name of an authorization rule when you use the Web Portal Manager or **pdadmin** commands.

- Ampersand (&)
- Asterisk (*)
- At sign (@)
- Backward slash (\)
- Colon (:)
- Comma (,)
- Double quotation (")
- Equal sign (=)
- Exclamation point (!)
- Left angle bracket (<)
- Left parenthesis (()
- Plus sign (+)
- Number sign (#)
- Right angle bracket (>)
- Right parenthesis ())
- Semicolon (;)
- Vertical bar (|)

A valid authorization rule name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Characters disallowed for ACL policy names

Certain characters are disallowed for ACL policy names.

These characters cannot be used in the name of an access control list (ACL) policy when you use the Web Portal Manager or **pdadmin** commands:

- Ampersand (&)
- Asterisk (*)
- At sign (@)
- Backward slash (\)
- Colon (:)
- Comma (,)
- Double quotation (")
- Equal sign (=)
- Exclamation point (!)
- Forward slash (/)
- Left angle bracket (<)
- Left parenthesis (()
- Period (.)
- Plus sign (+)
- Number sign (#)
- Right angle bracket (>)
- Right parenthesis ())
- Semicolon (;)
- Vertical bar (|)

A valid ACL policy name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Characters disallowed for POP names

Certain characters are disallowed for POP names.

Avoid the use of the following characters in the name of a protected object policy (POP) when you use the Web Portal Manager or **pdadmin** commands:

- Ampersand (&)
- Asterisk (*)
- At sign (@)
- Backward slash (\)
- Colon (:)
- Comma (,)
- Double quotation (")
- Equal sign (=)
- Exclamation point (!)
- Forward slash (/)
- Left angle bracket (<)
- Left parenthesis (()
- Period (.)

- Plus sign (+)
- Number sign (#)
- Right angle bracket (>)
- Right parenthesis ())
- Semicolon (;)
- Vertical bar (|)

A valid POP name is an alphanumeric string that is not case-sensitive. String values are expected to be characters that are part of the local code set. Spaces are not allowed.

Note: Although a POP name can contain 1 or more of these characters, the results of using such a POP are undefined.

Chapter 3. Using response files

A *response file* is a text file that contains product and system information, sometimes used in configuration.

Some utilities can be run in either command-line mode or response file mode.

- In command-line mode, all parameters must be specified from the command line.
- In response file mode, the utility obtains the necessary parameters from the response file. You must manually create the response file by entering all parameters.

Within response files, stanza labels display within brackets, such as `[stanza-name]`. Each stanza in a Security Verify Access response file contains one or more key value pairs. Key value pairs express information as a paired set of parameters. Each stanza entry is a key-value pair in the following format:

```
key = value
```

In the response file, the key is equal to the parameter in command-line mode. The following example shows that when there is a *`-parameter`* specified in command-line mode, the key in response file mode is the same, but without the preceding dash.

Examples

- The following example uses the `/tmp/rspfile/cars_pdacld.rsp` response file to configure an audit server by using SSL and password authentication:

  ```
  amauditcfg –rspfile /tmp/rspfile/cars_pdacld.rsp
  ```

  The `/tmp/rspfile/cars_pdacld.rsp` response file contains the following data:

  ```
  [amauditcfg]
  action = config
  srv_cfg_file = /opt/PolicyDirector/etc/ivacld.conf
  audit_srv_url = https://hostname:9443/CommonAuditService/services/Emitter
  enable_ssl = yes
  audit_key_file = /certs/WSclient.kdb
  audit_stash_file = /certs/WSclient.sth
  enable_pwd_auth = yes
  audit_id = administrator_id
  audit_pwd = password
  ```

- In contrast, the following example uses command-line mode to configure an audit server by using SSL and password authentication:

  ```
  amauditcfg -action config \
    -srv_cfg_file /opt/PolicyDirector/etc/ivacld.conf \
    -audit_srv_url https://hostname:9443/CommonAuditService/services/Emitter \
    -enable_ssl yes -audit_key_file /certs/WSclient.kdb \
    -audit_stash_file /certs/WSclient.sth -enable_pwd_auth yes \
    -audit_id administrator_id -auditpwd password
  ```

Index

IBM®