Secure, Role-Based Management of HP-UX Data Centers Using ServiceControl Manager

Richard D. Harrah
Hewlett-Packard Company
3404 East Harmony Road MS99-2UR9
Fort Collins CO 80528 - 9599
(970) 898 - 0012
richard_harrah@hp.com

1 Abstract

Administrators of UNIX data centers have traditionally grappled with their inability to provide anything other than "root or nothing" access to system and application administrators. This use model is inflexible and dated.

ServiceControl Manager enables effective administration of the HP-UX data center with secure, role-based administration of multiple HP-UX servers. Administrators are granted access to varying responsibilities on differing machines by virtue of the role they have been assigned by the ServiceControl Manager administrator – their authorizations.

2 Introduction

Traditional administration of multiple HP-UX servers from a central management console is usually accomplished with the remote shell (`r*`) commands. However, concerns around the inherent security problems of the `r*` commands are becoming commonplace. Subsequently, IT professionals require a safer method of centrally managing servers by executing a script or application on a remote machine, securely.

Methods to allow administrators to execute specific commands on a per-machine basis are available, and these include Access Control Lists, `sudo` and others. Unfortunately, these tools can be very cumbersome to manage on one machine, let alone a network of several machines with varying administrators and differing commands.

ServiceControl Manager provides the infrastructure to securely and robustly manage multiple HP-UX servers from a central location. This allows one administrator to not only manage more machines, but it also provides a single point – the Central Management Server – where a security policy can be configured and enforced.

One of the key benefits of ServiceControl Manager is the ability for a ServiceControl Manager Trusted User to partition the data center into groups of machines and groups of users that share an operation or function that is important in the data center – a *role*. Roles are also assigned to tools, and users are authorized logins on certain managed nodes in the data center. The result is that the security policy defines the tools administrators can execute on a managed node.

This paper is not a description of all aspects of ServiceControl Manager, and it is assumed that the reader is familiar with ServiceControl Manager to the extent that they understand the following primary objects: users (administrators), tools, roles, managed nodes and authorizations. General ServiceControl Manager concepts are described in “Reducing IT Management Costs Through the Service Control Manager” and an in-depth discussion on tools is presented in “ServiceControl Manager Tools,” both of which appear in these proceedings.

3 Roles

ServiceControl Manager roles allow groupings of administrators and tools along functional, organizational or any arbitrary criteria important to the requirements of the data center. These roles are also shared by ServiceControl Manager tools, mapping administrators to tools they can execute.

An example of a role is `troubleshooter`. Administrators assigned this role could be authorized to execute diagnostic or performance tools on nodes in the data center.

3.1 Attributes

The role is a fairly simple object, but its place in the ServiceControl Manager security policy is crucial for effective administration of the data center. Role attributes may be modified by the ServiceControl Manager Trusted User using the `mxrole(1M)` command. This command records role modification operations in the ServiceControl Manager central log.

3.1.1 Name

The role name is a unique identifier used to refer to the role. The role name can contain embedded spaces, dashes and underscores, and its length cannot exceed sixteen characters.

3.1.2 Role ID

The role ID is the unique identifier by which the ServiceControl Manager represents roles. This attribute is not exposed to the user and is used internally by the ServiceControl Manager.

3.1.3 Description

The description provides a field to contain a short line of text describing the role.

3.1.4 Enablement

ServiceControl Manager roles have the ability to be enabled or disabled. A tool that is assigned a disabled role prevents the tool from being executed by a user who has only that role on a specified node. An example of a situation where a role could be enabled or disabled is a role that a vendor’s field service engineer would use to run tools on machines when they are on site.

3.2 Semantics

Administrators who have similar responsibilities in the data center may be assigned a common ServiceControl Manager role. For example, administrators responsible for performing backup and restoration operations can be assigned a `backup ops` role. Then, all tools that accomplish backup and restoration functions can be assigned the `backup ops` role. The role links the user with the tool.

ServiceControl Manager provides a fixed set of sixteen (16) roles that can neither be deleted from, nor added to the ServiceControl Manager. With the exception of one role, all role names can be changed to allow the Trusted User to provide role names that make sense in the environment.

3.2.1 Master Role

The ServiceControl Manager role named `Master Role` is special in that its attributes cannot be modified. The `Master Role` allows a user assigned this role to run any tool on a any node assigned this role.

4 Authorizations

An authorization is the association between a user (administrator), role and node, and is the fundamental element of the security policy. Authorizations are created, deleted and listed from the GUI and with the `mxauth(1M)` command. All authorization operations are recorded in the ServiceControl Manager central log.

4.1 Attributes

4.1.1 User Name

The user name identifies the HP-UX login name of the user for the authorization. This user must be a ServiceControl Manager user.

4.1.2 Role Name

The role name identifies the name of the ServiceControl Manager role for the authorization.

4.1.3 Node Name

The node name identifies the Managed Node for which the user and role combination is valid. Future releases of ServiceControl Manager may allow authorizations to be specified for node groups as well.

4.1.4 Semantics

Authorizations are used by the ServiceControl Manager to determine if an administrator can execute a ServiceControl Manager tool on a node or group of nodes. A tool may be executed from the GUI or the command line. The administrator specifies the name of the tool to execute, along with any parameters for the tool, and the target nodes the tool is intended to execute on. This authorization for this tool invocation is carried out in the following steps:

1. Determine what ServiceControl Manager user is attempting to execute the tool
2. Determine which of the tool’s roles are enabled
3. Verify that the user is authorized a role on each target node

Tools can fail to execute for a number of reasons, but failures that occur because they violate the data center’s security policy are:

- The user has no authorizations
- The tool has no enabled roles
- There are no enabled roles shared by the tool and the user
- The user is not authorized a tool’s enabled role on a target node
- The user is not authorized to run any tools on a target node

Tool authorization fails as soon as a violation of the data center security policy is discovered. The nature of the tool failure is reported to the user and logged in the ServiceControl Manager central log.

5 Implementing a Security Policy

ServiceControl Manager enables very simple, as well as very robust and flexible, security policies to be configured and managed centrally from the Central Management Server. The complexity of the security policy is determined entirely by the ServiceControl Manager administrator.

A possible security policy configuration is one that aggregates administrators and machines into groups aligned on the functionality to be performed in the data center. After installing and initially configuring the ServiceControl Manager on the CMS, the following steps must take place to configure the security policy for the data center:

- Install the ServiceControl Manager agent software on each managed node
- Use the GUI or the command line to add managed nodes to the ServiceControl Manager repository
- Identify ServiceControl Manager administrators (users) and add them to the ServiceControl Manager with the GUI or the command line
- Add any environment-specific tools to the ServiceControl Manager repository with the GUI or the command line
- Configure node groups, using the GUI or the command line, comprised of nodes with common functionality
- Configure the roles and authorizations

5.1 A Simple Example

Assume that a customer has a data center comprised of a number of HP servers that are responsible for the following functions:

Business processes (payroll, billing, logistics, etc)
E-commerce (web servers)
Database applications for E-commerce

Also assume that there are operation/organizational roles

Backup/restore operations
HP customer/field support engineer

These may be just a subset of the roles that can be configured as the security policy. Further, once the roles are identified, the administrators responsible for those roles should be fairly identifiable.

5.1.1 Configure Roles

Roles may only be modified using `mxrole(1M)`, though they can be viewed with the GUI. Only ServiceControl Manager administrators with the Trusted User privilege are able to modify roles. ServiceControl Manager is initially configured with the following roles:

```
$ mxrole
Master Role
operator
dbadmin
webadmin
lvmadmin
role6
role7
role8
role9
```

```
role10
role11
role12
role13
role14
role15
role16
```

This form of `mxrole(1M)` simply lists the names of the roles. The Trusted User has determined that the following roles make sense in the data center:

- `operator`
- `bp ops`
- `backup ops`
- `webadmin`
- `dbadmin`
- `HP CE`

The `Master Role` is not modifiable, and the `operator` role is considered to be a valuable role in this data center. Now roles for the security policy may be modified in the following manner

```
$ mxrole –m role6 –N “bp ops” –d “ERP administrators”
$ mxrole –m lvmadmin –N “backup ops” –d “Role for backup/restore”
$ mxrole –m webadmin –d “Role for web server admin”
$ mxrole –m dbadmin –N “dbadmin” –d “Role for database admin”
$ mxrole –m role7 –N “HP CE” –d “HP field engineer” –e f
```

All roles are initially configured to be enabled. Note that the role for the HP field engineer, HP CE, is configured to be *disabled*. All role modification operations are recorded in the ServiceControl Manager central log.

5.1.2 Configure Authorizations

Now that roles are configured the authorizations may be created, thus defining the security policy for the data center. The operator role can reasonably be assumed to apply to all nodes in the data center. To configure the authorizations for administrators with this role, the following command is used

```
$ mxauth –a –u U –R operator –n ‘*’
```

where *U* is the login of the ServiceControl Manager administrator performing `operator` duties on **all** nodes within the data center. Therefore, user *U* on all nodes may execute all tools assigned the `operator` role. The wildcard is a convenience for stating that the user/role combination is to be defined for all ServiceControl Manager nodes.

Assume that the names of the nodes running the ERP software are named `erp`*`1`*` .. erp`*`N`*. Further assume that administrators `U`*`1`*` .. U`*`N`* are to be authorized to execute ERP-related tools on the `erp*` nodes. There may be a large number of authorizations for these nodes and administrators, but `mxauth(1M)` can only create or delete a single authorization on the command line. However, we can use the form of `mxauth(1M)` which enables multiple authorizations to be specified in a file for creation or deletion. The command is

```
$ mxauth –a –f /var/tmp/erp_auths
```

where each line of `erp_auths` is of the form

```
      username:rolename:nodename
```

This colon-delimited syntax fully defines all attributes for the authorization, and is effectively its *name*. The user name, role name and node name must be valid ServiceControl names. The contents of the file specifying authorization for our ERP administrators on the ERP nodes is

```
      U1:bp ops:erp1
      U1:bp ops:erp2
        . . .
        . . .
        . . .
      UN: bp ops:erpN
```

Again, the `mxauth(1M)` command logs a message in the ServiceControl Manager central log for each authorization created. We configure all authorizations in a similar manner. Once the security policy is in place, creating authorizations for new administrators or nodes is straightforward.

6 Conclusion

ServiceControl Manager allows administrators to manage the use and configuration of their machines by enforcing an easily configurable and very flexible security policy for their data center. Administrators are allowed to distribute and execute tools on authorized machines based on the roles in the data center. Therefore, ServiceControl Manager allows a data center administrator to configure and enforce a security policy for all ServiceControl Manager tools and administrators for all managed nodes in the data center.

An audit trail tracing the configuration of the security for the ServiceControl Manager is available in the ServiceControl Manager log file. All authorization creation and deletion, as well as role modifications appear in the log.

Appendix

This appendix contains preliminary man(1M) pages referenced in this paper.

NAME

mxrole - modify or list ServiceControl Manager roles

SYNOPSIS

```
mxrole -m rolename -N new_rolename
mxrole -m rolename -d description
mxrole -m rolename -e t|f
mxrole [-l n|t]
```

DESCRIPTION

The mxrole command allows a ServiceControl Manager (SCM) Trusted User to rename, describe and disable or enable SCM roles. SCM roles may not be added or removed. The Master Role is a special SCM role that may not be modified in any way.

The first form allows the user to modify the name of an SCM role.

The second form allows the user to modify the SCM-specific description for an SCM role. The existing description is replaced with the new description.

The third form allows the user to modify the enablement of the indicated SCM role. An SCM role enablement of t, or true, allows the execution of an SCM tool authorized by the SCM role.

The fourth form allows the user to list the SCM role names or to view the details of all SCM roles. When invoked with no options, mxrole gives a columnar listing of the names of all the SCM roles.

Only SCM Trusted Users may use mxrole to modify SCM roles. Any SCM user may display SCM role information.

Role Attributes

The following Role Attributes define an SCM role:

rolename   The name of the SCM role may have embedded spaces and its maximum length is 16 characters. SCM role names are not case sensitive.

Description The SCM-specific description for this SCM role. Its maximum length is 128 characters.

enablement   The state of enablement of the SCM role. A disabled SCM role prevents execution of any SCM tool on behalf of the SCM role.

Options

mxrole recognizes the following options:

-e t|f   Specifies that the SCM role enablement is to be set in the indicated manner - t (indicating enabled) or f (indicating disabled).

-l n   Produces a columnar list of the names of all SCM roles.

-l t   Indicates that a tabular display of SCM role attributes, for all SCM roles, is to be presented.

-m rolename Indicates the SCM role that is to be modified.

-N new_rolename

Defines the new SCM rolename for the specified SCM role.

-d description

Specifies the new SCM-specific description to be associated with the SCM role.

RETURN VALUE

mxrole returns one of the following values:

0	Successful completion.
1	Command line syntax error.
2	Error in a file operation.
5	Nonexistent role error.
21	Invalid name.
22	Invalid description.
23	Invalid ID.
24	Duplicate ID.
25	Role error.
26	Unsupported operation.
50	Unauthorized user.
102	SCM Repository error.
222	Central Management Server (CMS) is not initialized.
249	Unable to connect to the session manager.
250	Remote exception.
253	Duplicate name.

LIMITS

Valid role names may not exceed a length of 16 characters but may be as short as a single character. The initial character of a role name must be an upper- or lower-case letter.

Digits, underscores, dashes and spaces are legal role name elements. Leading and trailing white space is trimmed.

EXAMPLES

Disable the "HP CE" SCM role.

```
mxrole -m "HP CE" -e f
```

List the names of all SCM roles.

```
mxrole -l n
```

The output might look like the following:

```
    Master Role
    operator
    webadmin
    dbadmin
    lvmadmin
    role05
    role06
    role07
    role08
    role09
    role10
    role11
    role12
    role13
    role14
    role15
```

List the attributes for all SCM roles. SCM roles that have never been modified are shown as well.

```
    mxrole -l t
```

The output might look like the following:

```
    NAME        ENABLED? DESCRIPTION
    Master Role true     The SCM Master Role
    db admin    true     A role for db operations
    backup      true     A role for backup/restore operations
    HP CE       false    HP Field Service Engineer
    role4       true     For user by ServiceControl administrators
     ...
```

LIMITATIONS

This command may only be run on the CMS.

There is a limit of 16 SCM roles, of which all but the Master Role may be modified. The Master Role is a special SCM role that may not modified in any way, nor may this SCM role be removed from an SCM tool's SCM role authorizations. Roles may node be added or deleted.

AUTHOR

mxrole was developed by the Hewlett-Packard Company.

SEE ALSO

scmgr(1M), mxtool(1M).

NAME

mxauth - add, remove, or list authorizations in ServiceControl Manager

SYNOPSIS

```
mxauth -a -u username|UID -R rolename -n nodename
mxauth -a -f filename
mxauth -r -u username|UID -R rolename -n nodename
mxauth -r -f filename
mxauth [-l f|t]
```

DESCRIPTION

mxauth is used by a ServiceControl Manager (SCM) Trusted User to manage SCM authorizations. These associations between users, roles and nodes may only be added or removed by the SCM Trusted User, but any SCM user may list authorizations.

The first form of the command enables one authorization to be added. All options and associated parameters are required to completely specify an authorization. Asterisks are supported for the node parameter. An error results if any option or its data is missing.

The second form of the command allows multiple authorizations that are to be added, to be specified in a file. The authorizations must be in the Compact Authorization Format (see below).

The third form of the command enables one authorization to be removed. All options and associated parameters are required to completely specify an authorization. Asterisks are supported for the node parameter. An error results if any option or its data is missing.

As with adding authorizations, the authorization may be specified on the command line or in a file.

The last form of the command allows information about one or more authorizations to be listed.

When invoked with no options, mxauth lists all authorization names, in compact form, identical to the behavior of the -l f option.

qualifier provides a file-formatted listing. The output of the file-formatted listing is valid input to the -f option.

Compact Authorization Format

An authorization consists of a (user, role name, node name) association. The compact format for an authorization is:

```
user:rolename:nodename
```

This format provides a file-formatted listing. The output of the file-formatted listing is valid input to the -f option.

Options

mxauth recognizes the following options:

```
-a          Add authorization(s). The wildcard character (*)
            is supported for the node name argument.

-l  t       List all authorizations in the tabular format.

-l  f       List all authorization names in the Compact
            Authorization Format. This option may be used to
```

edit authorizations that have been saved to a file. This file may subsequently be used in conjunction with the -f option to modify authorizations.

-r Remove authorization(s). The wildcard character (*) is supported for the node name argument.

Authorization Attributes

-u username|UID
Specifies the user for the authorization.

-R rolename Specifies the role name for the authorization. Role names with embedded spaces must be enclosed in quotes.

-n nodename Specifies the node name for the authorization. The wildcard character (*) is supported for the node name argument.

RETURN VALUE

mxauth returns one of the following values:

0	Successful completion.
1	Command line syntax error.
2	Error in a file operation.
5	Nonexistent role error.
6	Nonexistent user error.
7	Nonexistent node error.
21	Invalid name.
25	Role error.
102	SCM Repository error.
222	Central Management Server (CMS) is not initialized.
249	Unable to connect to the session manager.
253	Duplicate name.

DIAGNOSTICS

mxauth writes to stdout, stderr and the SCM log file.

EXAMPLES

Add the authorizations defined in the file "my_auths".

```
mxauth -a -f my_auths
```

The contents of "my_auths" might look like:

```
joe:tester:*
martha:"db admin":chevss1
```

The first line above gives user "joe" the role of "tester" on all nodes in the ServiceControl Managed Cluster. The second line gives user "martha" the role of "db admin" on node "chevss1".

Remove authorization for user "martha" to execute any tools assigned to the "sapadmin" role on node "sap01".

```
mxauth -r -u martha -n sap01 -R sapadmin
```

List, in a tabular format, all roles assigned to all users in the ServiceControl Managed Cluster.

```
mxauth -l t
```

LIMITATIONS

This command may only be run on the CMS.

Removing Nonexistent Authorizations

It is not an error to remove an authorization that does not exist in the ServiceControl Manager. This operation results in an exit code of zero being returned by this command.

An authorization is nonexistent if all of its attributes (user name, role name or node name) are valid ServiceControl Manager, but they are not associated as an authorization.

Adding Duplicate Authorizations

It is not an error to add an authorization that already exists in the ServiceControl Manager. This operation results in an exit code of zero being returned by this command.

File Processing

If this command is being used to create ServiceControl Manager authorizations using the file-format form of this command, processing of the file halts when an invalid authorization is detected. Authorizations that have been added will be logged in the ServiceControl Manager log file and an error message indicating why the incorrect authorization is invalid is also recorded in the log file.

Conversely, if this command is being used to delete ServiceControl Manager authorizations using the file-format form and a nonexistent authorization is encountered, file processing is not halted because the nonexistent authorization is ignored.

AUTHOR

mxauth was developed by the Hewlett-Packard Company.

SEE ALSO

scmgr(1M), mxuser(1M), mxnode(1M), mxrole(1M).