Technical Report

Role-Based Access Control for Data ONTAP 7G

Ron Demery, NetApp
December 2009 | TR-3358

GRANULAR ADMINISTRATION OF CAPABILITIES

This paper is intended for storage administrators, security administrators, and IT management. It describes the role-based access controls (RBAC) introduced with Data ONTAP® 7G, an overview of the benefits of this feature, and some deployment examples.

TABLE OF CONTENTS

1 WHAT ARE ROLE-BASED ACCESS CONTROLS? 3

2 HOW DOES RBAC WORK IN DATA ONTAP? 3

2.1 USERS 3

2.2 GROUPS 4

2.3 ROLES 4

2.4 CAPABILITIES 4

2.5 PULLING IT ALL TOGETHER 6

3 INTEGRATION WITH MICROSOFT ACTIVE DIRECTORY 6

4 CAPABILITIES 7

4.1 APPLICATION PROGRAMMING INTERFACE (API) CAPABILITIES 7

4.2 COMMAND LINE INTERFACE (CLI) CAPABILITIES 9

5 DEPLOYMENT EXAMPLES 13

5.1 CREATING AN SNMP ADMINISTRATOR 14

5.2 CREATING A USER WHO ONLY MAKES SNMP REQUESTS 14

5.3 CREATING/MODIFYING A USER TO NOT HAVE CONSOLE ACCESS 14

5.4 CUSTOMIZING ACCESS THROUGH SYSTEM MANAGER 15

6 CONCLUSION 17

1 WHAT ARE ROLE-BASED ACCESS CONTROLS?

Role-based access controls, or RBAC, are a method for managing the set of actions that a user or administrator may perform in a computing environment.

Historically, older computer operating systems allowed any user who had access to the system to perform any function. In fact, many systems did not distinguish between users at all. Most current operating systems provide, at a minimum, the ability to create several different users, each with a separate username and password. Once the ability to distinguish between users was provided, operating systems began to use user identification as a means to control access to files, directories, and other system objects. Good examples of this are the file permissions used on UNIX® systems (and the NFS protocol) or the access control lists (ACLs) used on Windows® systems (and the CIFS protocol).

In addition to file access, there are other actions that should be managed for security reasons. For example, only the system administrator should be allowed to add new user accounts to the system. From this it becomes clear that the users who access a system fall into at least two categories, or roles: administrators and nonadministrators.

While reserving certain functions for administrator-only access is a good start, additional problems need to be solved. Most organizations have multiple system administrators, some of whom require more privileges than others. By selectively granting or revoking privileges for each user, you can customize the degree of access that an administrator has to the system. As one example, Microsoft® Windows provides this capability. The problem with this approach is that as the number of system administrators grows, it becomes difficult and time consuming to manage the set of capabilities granted to each administrator.

Role-based access controls solve this management problem by allowing you to define sets of capabilities (roles) that are not assigned to any particular user. Users are assigned to groups based on their job functions, and each group is granted the set of roles required to perform those functions. Using this method, the only configuration required for an individual administrator is to make sure that that administrator is a member of the appropriate groups; that administrator will inherit all the correct capabilities because of the group membership and the roles assigned to those groups.

2 HOW DOES RBAC WORK IN DATA ONTAP?

While the overall concept of role-based access controls is applicable to a wide range of operating systems and applications, the details of how RBAC is implemented vary depending on the OS or application in use. This section describes the specific terminology and architecture used in Data ONTAP; it is important to understand these concepts and definitions before configuring RBAC in Data ONTAP, especially if you have experience with RBAC implementations in other software (because the terminology or architecture might be different from implementations you have used in the past).

2.1 USERS

A user is defined as an account that is authenticated on the NetApp® system.

A domain user is defined as a nonlocal user who belongs to a Windows domain and is authenticated by the domain.

Both users and domain users represent individual, authenticated humans. While it is possible to define a user or domain user that represents a piece of software or that is shared among multiple humans, this is not the most common scenario and is not discussed in depth here.

Both users and domain users, as discussed in this document, are assumed to be authorized system administrators. Normal, nonadministrative users who access files on the system using CIFS or NFS, or who use client systems that mount LUNs using FCP or iSCSI, are not discussed in this document. They have no ability to log into or manage a Data ONTAP system unless they have been specifically defined as either users or domain users with the useradmin command.

Refer to the “How to Manage Users” section of the *System Administration Guide* for further information.

2.2 GROUPS

A group is defined as a collection of users and/or domain users. Groups may be assigned one or more roles. It is important to remember that the groups defined within Data ONTAP are separate from the groups defined in other contexts, such as a Microsoft Active Directory server. This is true even if the groups within Data ONTAP have the same names as groups elsewhere within your environment.

When creating new users and/or domain users, Data ONTAP requires specification of group membership. Therefore, it is best to create appropriate groups before defining users or domain users.

The default groups are administrators, backup operators, compliance administrators, guests, power users, and users.

Refer to the "How to Manage Groups" section of the *System Administration Guide* for further information.

2.3 ROLES

A role is defined as a named set of capabilities. Data ONTAP comes with several roles predefined, and users may create additional roles or modify the provided roles.

The default roles are admin, audit, backup, compliance, none, power, and root.

Refer to the "How to Manage Roles" section of the *System Administration Guide* for further information.

2.4 CAPABILITIES

A capability is defined as the privilege granted to a role to execute commands or take other specified actions.

Table 1) Capability types.

Capability Type	Description
`api`	Grants the specified role the capability to execute Data ONTAP API calls. The `api-*` type includes all of the Data ONTAP API calls. These commands are only available with `login-http-admin`, so in general, any `api-*` command must also include this login. The format for this is `api-<ontap-api-command>`, which means allow a specific command/subcommand. Here, it is possible to list only subcommands, like `api-system-get-info`, or a command and its subcommands, like `api-system-get-*` , or even `api-system-*`. `api-*` Grants the specified role all api capabilities. `api-api_call_family-*` Grants the specified role the capability to call all API routine in the family api_call_family. `api-api_call` Grants the specified role the capability to call the API routine api_call. Note: You have more fine-grained control of the command set with the api capabilities because you can give subcommand capabilities as well. Users with api capability also require the login-http-admin capability to execute API calls.

Capability Type	Description
`cli`	Grants the specified role the ability to execute one or more Data ONTAP command line interface (CLI) commands. The `cli-*` category includes all of the commands that can be run after a user is logged in with telnet, console, rsh, or ssh. The format for this is `cli-<command>*` , which means allow all the commands and subcommands. (`cli-<command>` just means the command and NO subcommands.) The capability for a specific command, like `exportfs`, would have the following syntax: `cli-exportfs*`. This means allow command line accesses to the `exportfs` command and all of its subcommands. `cli-export*` might look valid but is NOT allowed. `cli-*` Grants the specified role the capability to execute all supported CLI commands. `cli-cmd*` Gives the specified role the capability to execute all commands associated with the CLI command cmd. For example, the following command gives the specified role the capability to execute all vol commands: `useradmin role modify status_gatherer -a cli-vol*` Note: Users with cli capability also require at least one login capability to execute CLI commands.
`compliance`	Grants the specified role the ability to execute compliance-related operations. `compliance-*` Grants the specified role the capability to execute all compliance-related operations. `compliance-privileged-delete` Grants the specified role the capability to execute privileged deletion of compliance data. Note: The compliance capabilities (compliance-*) are included in the default capabilities of the compliance role. The compliance capabilities cannot be removed from the compliance role or added to other roles.
`filerview`	Grants the specified role read-only access to FilerView®. This capability type includes only the `filerview-readonly` capability, which grants the specified role the capability to view but not change manageable objects on systems managed by FilerView. Note: There is no predefined role or group for read-only FilerView access. You must first assign the filerview-readonly capability to a role and then assign the role to a group, before you can create a user in such a group.
`login`	Grants the specified role telnet, console, rsh, ssh, or http-admin login capabilities. `login-*` Gives the specified role the ability to log in through all supported protocols. login-protocol Gives the specified role capability to log in through a specified protocol. Supported protocols include: `login-telnet` Gives the specified role the ability to log in to the storage system using Telnet. `login-console` Gives the specified role the ability to log in to the storage system using the console. `login-rsh` Gives the specified role the ability to log in to the storage system using rsh. `login-ssh` Gives the specified role the ability to log in to the storage system using SSH. `login-http-admin` Gives the specified role the ability to log in to the storage system using HTTP. `login-snmp` Gives the specified role the ability to log in to the storage system using SNMPv3. `login-ndmp` Gives the specified role the ability to make NDMP requests.

Capability Type	Description
`security`	Grants the specified role security-related capabilities, such as the ability to change other users' passwords or to invoke the CLI priv set advanced command. `security-*` Grants the specified role all security capabilities. `security-`*`capability`* Grants the specified role one of the following specific security capabilities: `security-api-vfiler` Normally a client will send Data ONTAP APIs directly to a vFiler™ unit if it wants the API to be executed on the vFiler unit. The security-api-vfiler capability is necessary to send Data ONTAP APIs to the physical storage system which are to be forwarded to a vFiler unit for execution. By default, only root and members of the administrators group have this capability. `security-passwd-change-others` Gives the specified role the capability to change the passwords of all users with equal or less capabilities. By default, only root and members of the administrators group have this capability. `security-priv-advanced` Gives the specified role the capability to access the advanced CLI commands. This is necessary to run advanced commands that are not used for normal administration. Please talk to a NetApp representative before using advanced commands. By default, only root and members of the administrators group have this capability. `security-load-lclgroups` Gives the specified role the capability to reload the lclgroups.cfg file. By default, only root and members of the administrators group have this capability. `security-complete-user-control` Gives the specified role the capability to create, modify, and delete users, groups, and roles with greater capabilities. These users typically only have access to the `cli-useradmin*` and associated commands, though they can give themselves greater permissions. By default, only root and members of the administrators group have this capability.

2.5 PULLING IT ALL TOGETHER

Users are members of groups, groups have one or more roles, and each role grants a set of capabilities. In this way Data ONTAP allows you to create flexible security policies that match your organizational needs.

All configuration for role-based access controls occurs with the `useradmin` command provided by Data ONTAP. For example, users are added or modified with the `useradmin user add` or `useradmin user modify` command. This section includes specific command-line instructions from the *Data ONTAP System Administration Guide* for your convenience. Example deployment scenarios are provided in section 5 of this paper.

Because users and domain users must be members of groups, and because groups must be assigned one or more roles, the best sequence for configuration tasks is to create the roles first; then create groups and assign roles to them; and finally create users and domain users, providing them with appropriate group membership.

Detailed documentation on how to use the `useradmin` command to define users, domain users, groups, and roles is provided in the *Data ONTAP System Administration Guide*. This paper is not intended to replace the documentation in the *System Administration Guide*.

3 INTEGRATION WITH MICROSOFT ACTIVE DIRECTORY

The ability to define domain users that are authenticated by an Active Directory domain rather than Data ONTAP is a powerful tool for managing large storage environments. Most enterprise computing environments already have an Active Directory infrastructure available, and storage administrators or other users who need administrative access to storage devices already have accounts defined within that Active Directory infrastructure. By using this preexisting authentication capability, rather than defining separate accounts for the storage environment, several key benefits are obtained:

- An administrator's authentication credentials (username, password) are the same when logging into the storage system as they are when logging into any Windows system in the environment. When the password is changed within the Windows environment, the change immediately takes effect within the storage environment.
- Additionally, changing an administrator's password once, within Active Directory, has the effect of changing it on all storage devices to which that administrator has access. This is a significant reduction in management overhead for environments with a large number of storage devices.
- Centralized authentication allows local security policy, implemented within Active Directory, to take effect across all storage devices as well. For example, administrators might be compelled to change their passwords with a certain frequency and might be provided with advance warning as password expiration time approaches. Likewise, when they do change passwords, the Active Directory environment can enforce policy regarding password composition and length checking, reuse of previous passwords, use of dictionary words in passwords, and so on.
- When an administrator departs from an organization, disabling that administrator's Active Directory account has the side effect of immediately revoking access to the storage environment as well.

However, it would be inadvisable to provide *all* of the accounts within Active Directory access to storage management functions. Obviously only a subset of the AD accounts represents administrative staff, and only a subset of the administrative staff (in any large organization) will have a need to administer storage controller systems. Any system that provides transparent Active Directory authentication on a storage system without discriminating between authorized administrators and other accounts will be exposing the storage system to huge security problems.

To avoid such problems, Data ONTAP only authenticates an administrator against Active Directory if that administrator has been defined as a domain user using the useradmin command.

To maximize the benefit of this new capability, consider migrating storage system and NearStore® administrator accounts from local users to domain users. This is a simple process:

- Make sure that the storage devices have been upgraded to Data ONTAP 7G or later.
- Use the *useradmin user delete* command to delete the local usernames and passwords.
- Use the *useradmin domainuser add* command to grant authorized Active Directory accounts administrator access.

4 CAPABILITIES

The capabilities associated with a role can be defined as through the cli or through the api capabilities. The capabilities will differ between versions of Data ONTAP.

4.1 APPLICATION PROGRAMMING INTERFACE (API) CAPABILITIES

This listing is taken from the Manage ONTAP® SDK 3.5.1. Further information concerning the Manage ONTAP SDK can be found on the NOW™ site.

An example to assign the aggr access for the api interface would be:

```
filer*> useradmin role add newrole –a login-http-admin,api-aggr*
```

The Data ONTAP APIs are used to access and manage the NetApp storage system. These are a proprietary set of APIs. This set includes APIs for security management, license management, backup and recovery, data replication, data archiving, and so on.

Table 2) API capability and description.

api Capability	Description
`aggr`	Provides aggregate information and management
`cf`	Provides the cluster failover operations
`cifs`	Provides the CIFS setup, sessions, and shares
`clock`	Provides system time zone and date management

api Capability	Description
clone	Manages file and subfile cloning operation
consistency	Provides consistency group management
dfm	DataFabric® Manager server-side settings
disk	Provides disk-related operations
ems	Provides APIs to perform event management system (EMS)
fc	Provides Fibre Channel adapter configuration
fcp	Provides Fibre Channel protocol management
fcpport	Provides APIs that manage the Fibre Channel protocol
file	Provides APIs that support file system operations
fpolicy	Provides file policy and configures file management
ic	Provides APIs that do storage failover interconnect
igroup	Provides APIs that do initiator group operations
ipspace	Provides ipspace management APIs
iscsi	Provides iSCSI management and monitoring
license	Provides license code management
lock	Provides lock manager configuration and management
lun	Provides APIs that manage and monitor the LUNs
nameservice	Provides APIs that obtain name service information
net	Provides access to Data ONTAP network management
nfs	NFS configuration and management
options	Provides option values management
perf	Provides counters of various system performance objects management
portset	Provides port set operations
priority	Provides APIs that manage priority scheduling operations
qtree	Provides qtree management
quota	Provides APIs for quota editing and management
reallocate	Provides LUN management and file optimization, and reallocation operations
rsh	Provides rsh information
ses	Provides SCSI enclosure services-related operations
sis	Provides advanced single-instance storage management
snaplock	Provides API SnapLock® information and management
snapmirror	Provides SnapMirror® management
snapshot	Manages Snapshot™ copies
snapvault	Provides SnapVault® management
snmp	Provides SNMP management and retrieving the MIB values
software	Provides software package information and management
storage-adapter	Provides storage adapter operations
system	Provides APIs that retrieve system information

api Capability	Description
useradmin	Provides APIs that configure and manage user administrator
vfiler	Provides vFiler unit configuration and management
volume	Provides volume information and management
wafl	Provides APIs that do utility functions for agents

4.2 COMMAND LINE INTERFACE (CLI) CAPABILITIES

This list of capabilities was derived from the man pages from the *Data ONTAP 7.3 Commands: Manual Page Reference, Volume 1.*

An example to assign the aggr access for the cli interface would be:

```
filer*> useradmin role add newrole –a login-ssh,cli-aggr*
```

Table 3) CLI capabilities and descriptions.

cli Capability	Description
acpadmin	Commands for managing alternate control path administrator
aggr	Commands for managing aggregates, displaying aggregate status, and copying aggregates
arp	Address resolution display and control
backup	Manages backups
bmc	Commands for use with a baseboard management controller (BMC)
bootfs	Boot file system access command (advanced)
charmap	Command for managing per-volume character maps
cf	Controls the takeover and giveback operations of the storage systems in a cluster
cifs-*	Commands for managing cifs
cifs-access	Modify share-level access control or Windows machine account access
cifs-adupdate	Update the storage system's account information on the Active Directory server
cifs-audit	Configure CIFS auditing
cifs-broadcast	Display a message on user workstations
cifs-changefilerpwd	Schedules a domain password change for the storage system
cifs-comment	Display or change CIFS server description
cifs-domaininfo	Display domain type information
cifs-help	Display help for CIFS-specific commands
cifs-homedir	Manage CIFS home directory paths
cifs-lookup	Translate name into SID or vice versa
cifs-nbalias	Manage CIFS NetBIOS aliases
cifs-prefdc	Configure and display CIFS preferred domain controller information
cifs-resetdc	Reset CIFS connection to domain controller
cifs-restart	Restart CIFS service
cifs-sessions	Information on current CIFS activity
cifs-setup	Configure CIFS service
cifs-shares	Configure and display CIFS shares information

cli Capability	Description
cifs-sidcache	Clears the CIFS SID-to-name map cache
cifs-stat	Print CIFS operating statistics
cifs-terminate	Terminate CIFS service
cifs-testdc	Test the storage system's connection to Windows NT® domain controllers
cifs-top	Display CIFS clients based on activity
clone	Manages file and subfile cloning
config	Command for configuration management
date	Display or set date and time
dd	Copy blocks of data
df	Display free disk space
disk	RAID disk configuration control commands
disk_fw_update	Update disk firmware
disktest	Disk test environment
dlm	Administer dynamically loadable modules
dns	Display DNS information and control DNS subsystem
download	Install new version of Data ONTAP
dump	File system backup
echo	Display command line arguments
ems	Invoke commands to the Data ONTAP event management system
environment	Display information about the storage system's physical environment
exportfs	Exports or unexports a file system path, making it available or unavailable, respectively, for mounting by NFS clients
fcadmin	Commands for managing Fibre Channel adapters
fcdiag	Diagnostic to assist in determining source of loop instability
fcp	Commands for managing Fibre Channel target adapters and the FCP target protocol
fcstat	Fibre Channel stats functions
fctest	Test Fibre Channel environment
file	Manage individual files
filestats	Collect file usage statistics
flexcache	Commands for administering FlexCache® volumes
floppyboot	Describes the menu choices at the floppy boot prompt
fpolicy	Configure file policies
fsecurity	Summary of fsecurity commands
fsecurity-apply	Creates a security job based on a definition file and applies it to the file system
fsecurity-cancel	Cancels outstanding fsecurity jobs
fsecurity-help	Displays a description and usage information for fsecurity commands
fsecurity-remove-guard	Removes the storage-level access guard from a volume or qtree
fsecurity-show	Displays the security settings on files and directories

cli Capability	Description
fsecurity-status	Displays the status of outstanding fsecurity jobs
ftp	Display FTP statistics
ftpd	File transfer protocol daemon
halt	Stop the storage system
help	Print summary of commands and help strings
hostname	Set or display storage system name
httpstat	Display HTTP statistics
ifconfig	Configure network interface parameters
ifinfo	Display driver-level statistics for network interfaces
ifstat	Display device-level statistics for network interfaces
igroup	Commands for managing initiator groups
ipsec	Manipulates the ipsec SP/SA/certificate databases and displays ipsec statistics
ipspace	ipspace operations
iscsi	Manage iSCSI service
iswt	Manage the iSCSI software target (ISWT) driver
keymgr	Key and certificate management
license	License Data ONTAP services
lock	Manage lock records
logger	Record message in system logs
logout	Allows a user to terminate a telnet session
lun	Commands for managing luns
man	Locate and display reference manual pages
maxfiles	Increase the number of files the volume can hold
memerr	Print memory errors
mt	Magnetic tape positioning and control
nbtstat	Displays information about the NetBIOS over TCP connection
ndmpcopy	Transfers directory trees between storage systems using NDMP
ndmpd	Manages NDMP service
ndp	Control/diagnose IPv6 neighbor discovery protocol
netdiag	Perform network diagnostics
netstat	Show network status
nfs	Turn NFS service off and on, or set up Kerberos V5 for NFS
nfsstat	Display NFS statistics
nis	Display NIS information
options	Display or set storage system options
orouted	Old network routing daemon
partner	Access the data on the partner in takeover mode
passwd	Modify the system administrative user's password

cli Capability	Description
ping	Send ICMP ECHO_REQUEST packets to network hosts
ping6	Send ICMPv6 ECHO_REQUEST packets to network hosts
pktt	Controls on storage system packet tracing
portset	Commands for managing portsets
priority	Commands for managing priority resources
priv	Control per-connection privilege settings
qtree	Create and manage qtrees
quota	Control storage system disk quotas
rdate	Set system date from a remote host
rdfile	Read a WAFL® file
reallocate	Command managing reallocation of files, LUNs, volumes, and aggregates
reboot	Stop and then restart the storage system
restore	File system restore
rlm	Commands for use with a remote LAN module (RLM)
rmc	Commands for use with a remote management controller
route	Manually manipulate the routing table
routed	Network RIP and router discovery routing daemon
rshstat	Prints the information about active rsh sessions
rtsold	Router solicitation daemon
san	Glossary for NetApp specific SAN terms
sasadmin	Commands for managing Serial Attached SCSI (SAS) adapters
sasstat	Commands for managing Serial Attached SCSI (SAS) adapters
savecore	Save a core dump
sectrace	Manages permission tracing filters
secureadmin	Command for secure administration of the appliance
setup	Update storage system configuration
sftp	Display SFTP (SSH File Transfer Protocol) statistics
shelfchk	Verify the communication of environmental information between disk shelves and the storage system
sis	Advanced single-instance storage (SIS) management
snap	Manage Snapshot copies
snaplock	Compliance related operations
snapmirror	Volume, and qtree mirroring
snapvault	Disk-based data protection
snmp	Set and query SNMP agent variables
software	Command for install/upgrade of Data ONTAP
source	Read and execute a file of storage system commands
stats	Command for collecting and viewing statistical information

cli Capability	Description
`storage`	Commands for managing the disks and SCSI and Fibre Channel adapters in the storage subsystem
`sysconfig`	Display storage system configuration information
`sysstat`	Report storage system performance statistics
`timezone`	Set and obtain the local time zone
`traceroute`	Print the route packets take to network host
`traceroute6`	Print the route IPv6 packets take to a network node
`ups`	Controls the monitoring of uninterruptible power supplies (UPSs)
`uptime`	Show how long system has been up
`useradmin`	Administer storage system access controls
`version`	Display Data ONTAP version
`vfiler`	vFiler unit operations
`vif`	Manage virtual network interface configuration
`vlan`	Manage VLAN interface configuration
`vol`	Commands for managing volumes, displaying volume status, and copying volumes
`vscan`	Control virus scanning for files on the storage system
`wcc`	Manage WAFL credential cache
`wrfile`	Write a WAFL file
`ypcat`	Print values from a NIS database
`ypgroup`	Display the group file entries cached locally from the NIS server if NIS is enabled
`ypmatch`	Print matching values from a NIS database
`ypwhich`	Display the NIS server if NIS is enabled

5 DEPLOYMENT EXAMPLES

Data ONTAP role-based access controls have the flexibility to meet the needs of almost any IT environment. How they are used will depend largely on local security policies and organizational structure. The following are just a few examples of how RBAC might be used to enhance security and manageability in an enterprise IT environment.

USING THE `useradmin role` COMMAND

```
useradmin role add role_name  [-c  comments]  -a
capability1[,capability2,...,capabilityN]
```

```
useradmin role modify role_name [-c comments] [-a capabil-
ity1,capability2,...,capabilityN]
```

`role add` and `role modify` are used to add and modify administrative roles. The role name has all the restrictions of a user name.

The -a option specifies which capabilities are allowed in this role. This option completely replaces this role's current capabilities with the new ones.

The -c option specifies a comment about the role. Comments for roles have all the restrictions of user comments.

```
useradmin role delete role_name
```

`role delete` is used to delete an administrative role.

```
useradmin role list [role_name ]
```

`role list` is used to list administrative roles. Giving a role name just lists a single role. The role entries will each be printed in list format as follows:

```
Name:    none
Info:    Default role for no privileges.
Allowed Capabilities:
```

(The above indicates the role "none" does not have any capabilities.)

```
Name:    power
Info:    Default role for power user privileges.
Allowed Capabilities: cli-cifs*,cli-exportfs*,cli-nfs*,cli-useradmin*,api-cifs-
*,api-nfs-*,login-telnet,login-http-admin,login-rsh,login-ssh,api-system-api-*
```

5.1 CREATING AN SNMP ADMINISTRATOR

This creates two roles, one which can rsh into the storage system and run the help command, and another which is allowed to log in through any login method and run any SNMP command. The "snmp_admins" group is allowed to log into the storage system and run the help command through telnet, rsh, SNMPv3, and so on and make get and get next requests. The user "wilma" inherits these capabilities from the group.

```
useradmin role add rsh_help -a login-rsh,cli-help*
useradmin role add snmp_commands -a login-*,cli-snmp*,api-snmp-*
useradmin group add snmp_admins -r rsh_help,snmp_commands
useradmin user add wilma -g snmp_admins
```

5.2 CREATING A USER WHO ONLY MAKES SNMP REQUESTS

This creates a role and group whose only capability is making SNMP requests. The storeMgr client inherits this capability.

```
useradmin role add snmp_requests -a login-snmp
useradmin group add snmp_managers -r snmp_requests
useradmin user add storeMgr -g snmp_managers
```

5.3 CREATING/MODIFYING A USER TO NOT HAVE CONSOLE ACCESS

A user without console access cannot execute any storage system CLI commands. These local users should be placed in local groups that do not have any roles which contain these capabilities. To see if a user has access, list the user and check the allowed capabilities. If a user is in a group with the capabilities "`cli-*`" and "`login-*`", then that user has console access. The following command places a user into a group with no capabilities, which will revoke all privileges.

```
useradmin user modify myuser -g "Guests"
useradmin user list myuser
```

5.4 CUSTOMIZING ACCESS THROUGH SYSTEM MANAGER

Netapp System Manager is a GUI-based storage controller administration tool. There might be requirements to allow a “view-only” administrator to have access using this tool. In order to accomplish this, it is necessary to develop custom roles for logon and view-only access. This example contained the required capabilities and roles to enable a “view-only” account for the “storage” container in the NetApp System Manager tree.

Note: This example was developed using NetApp System Manager 1.0.1 and Data ONTAP 7.3.2.

CREATE THE ROLES

Seven roles are created in this example; it is possible to have only one role with all of the capabilities. The breakdown of roles was done for simplicity.

Create a login role. This role allows the access to the storage controller:

```
useradmin role add nsm-login -a login-http-admin,api-system-get-*
```

Create a common “view-only” role. This role has all of the common capabilities to access the “storage” container of the NetApp System Manager GUI:

```
useradmin role add nsm-view –a api-aggr-list-info,api-disk-sanown-list-info,api-license-list-info,api-options-get,api-perf-object-get-instances,api-snmp-status,api-volume-list-info*,cli-priv,api-aggr-options-list-info,api-aggr-check-spare-low
```

Create a role to vew the “volumes” container:

```
useradmin role add nsm-volumes-view -a api-volume-get-root-name,api-snapshot-reserve-list-info,api-volume-get-language,api-volume-options-list-info,cli-date
```

Create a role for the “shared folders” container:

```
useradmin role add nsm-sharedfolders-view -a api-cifs-share-list-iter*,api-nfs-exportfs-list-rules,api-cifs-session-list-iter*
```

Create a role to view the “qtrees” container:

```
useradmin role add nsm-qtree-view -a api-qtree-list-iter*
```

Create a role to view the “disks” container:

```
useradmin role add nsm-disk-view -a api-system-cli,api-disk-list-info,cli-options
```

Create a role to view the “aggregates” container:

```
useradmin role add nsm-aggr-view -a api-aggr-get-root-name,api-snapshot-list-info
```

The output of the `useradmin role list` command should contain the following:

Name: nsm-aggr-view

Info:

Allowed Capabilities: api-aggr-get-root-name,api-snapshot-list-info

Name: nsm-disk-view

Info:

Allowed Capabilities: api-system-cli,api-disk-list-info,cli-options

Name: nsm-login

Info:

Allowed Capabilities: login-http-admin,api-system-get-*

Name: nsm-qtree-view

Info:

Allowed Capabilities: api-qtree-list-iter*

Name: nsm-sharedfolders-view

Info:

Allowed Capabilities: api-cifs-share-list-iter*,api-nfs-exportfs-list-rules,api-cifs-session-list-iter*

Name: nsm-view

Info:

Allowed Capabilities: api-aggr-check-spare-low,api-aggr-list-info,api-aggr-options-list-info,api-disk-sanown-list-info,api-license-list-info,api-options-get,api-perf-object-get-instances,api-snmp-status,api-volume-list-info*,cli-priv,security-priv-advanced,cli-registry

Name: nsm-volumes-view

Info:

Allowed Capabilities: api-volume-get-root-name,api-snapshot-reserve-list-info,api-volume-get-language,api-volume-options-list-info,cli-date

CREATE THE GROUP

Once the roles have been created, a group needs to be created to allow the acess to all areas of the “storage container” for the view-only account.

```
useradmin group add nsm-storage-view -r nsm-login,nsm-view,nsm-volumes-view,nsm-sharedfolders-view,nsm-qtree-view,nsm-disk-view,nsm-aggr-view
```

The minimum roles for the “view-only” group are the nsm-login and the nsm-view roles. These roles will provide some access of information.

CREATE OR ADD USERS TO THE GROUP

This is accomplished through either the useradmin user or the useradmin domainuser commands.

6 CONCLUSION

Role-based access controls in Data ONTAP allow storage and security administrators to map administrative capabilities to local security policy. This can assist in meeting regulatory or internal policy requirements, can protect in many cases against accidental misconfigurations and other user errors, and can allow storage administrators to delegate responsibility for certain tasks without providing untrained administrators with access to destructive commands.