Managing the Network

This chapter details the options available to the system administrator to manage the network by easily adding devices into Cisco E-DI and grouping them for operational use:

- Credential sets—Specify how to communicate with the managed devices.
- Discovery—Devices need to be discovered before they are managed.
- Static and dynamic device grouping—Provides context for the Cisco E-DI CLI operations.
- Interface grouping—A set of static system-defined groups that combine multiple network interfaces into a single interface which may be used for configuring several interfaces at once.

Cisco E-DI provides session based device authentication for networks where there is an external AAA server. This mode requires a user to enter a login and password when managing devices. See Device Authentication and Using Session Based Device Authentication for more information. Session based device authentication is disabled by default, and must be enabled before any devices are managed. This can be done by the system administrator during installation, or by entering the following command in server configuration mode:

```
[SVR:/server](config)# device-auth session-based
```

To specify the session credentials, enter:

```
[SVR:/server]# terminal device-auth login <login val>
```

This chapter includes the following information:

Creating Credential Sets

Device credentials like login, password, and SNMP community string settings are required for communication with a device. Cisco E-DI combines these credentials into a credential set which specifies the necessary information for Cisco E-DI to communicate to the device. It is assigned to a device when the device is managed. See Chapter 1, "Cisco E-DI Concepts" for more information about credential sets.

The commands used to create the credential sets are detailed in Table 3-1. The commands are given in server configure credential set mode `[SVR:/server](conf-credential-set)#`.

Table 3-1 Commands to Create Credential Sets

Action	Command
Enter credential configuration mode by specifying a credential set name to configure or assign attributes to the default credential set.	`[SVR:/server](config)# credential-set {default | name}`
A new credential set can be created based on an existing credential set. The new credential set inherits the attributes of the existing credential set.	`[SVR:/server] (config)# credential-set [new name] based-on [name]`
To select Telnet transport.	`[SVR:/server](conf-credential-set)# transport telnet`
To select SSH v1.5 transport. **Note** The SSH default is 3des.	`[SVR:/server](conf-credential-set)# transport ssh [cipher] { 3des | aes_128 | aes_192 | aes_256 | blowfish | des | twofish }`
To select SSH v2 transport. **Note** The SSH default is 3des. Modes are applicable for all ciphers except arcfour.	`[SVR:/server](conf-credential-set)# transport ssh2 [cipher] { 3des | aes_128 | aes_192 | aes_256 | arcfour | blowfish | twofish_128 | twofish_192 | twofish_256 } [mode] { cbc | cfb | ctr | ecb | ofb |}`

To specify the enable password login for Telnet.
Note The behavior of these commands changes when session based device authentication is enabled. See Using Session Based Device Authentication, page 2-6 for a full explanantion of the command behavior.
	`[SVR:/server](conf-credential-set)# enable-password [{<0-2> name} | name]`
	`[SVR:/server](conf-credential-set)# password [{<0-2> name} | name]`
	`[SVR:/server](conf-credential-set)# login [{<0-2> name} | name]`

To specify the read community for SNMP communication.	`[SVR:/server](conf-credential-set)# read-community [{<0-2> name } | name]`
To specify the write community for SNMP communication.	`[SVR:/server](conf-credential-set)# write-community [{<0-2> name} | name]`

Table 3-1 Commands to Create Credential Sets (continued)

Action	Command
To remove a credential set.	`[SVR:/server](conf)# no credential-set name`

To set the value of a command to null, use **no** before the command.	`[SVR:/server](conf-credential-set)# no read-community`
	
	`[SVR:/server](conf-credential-set)# no write-community`
	
	`[SVR:/server](conf-credential-set)# no login`
	
	`[SVR:/server](conf-credential-set)# no password`
	
	`[SVR:/server](conf-credential-set)# no enable-password`
	
	`[SVR:/server](conf-credential-set)# no transport`

The following example shows two credential sets:

```
credential-set default
read-community 2 681D7F137A19
write-community 2 681D7F137A19
login Cisco
password 2 573E4D2E41
enable-password 2 286B0271127D
transport telnet

credential-set Switch
read-community 2 681D7F137A19
write-community 2 681D7F137A19
login switch
password 2 7F127719
enable-password 2 7F127719
transport telnet
```

Sample credential set created using the **based-on** option:

```
credential-set <new name> based-on <name>
transport ssh
```

The credential set <new name> has all the attributes of the credential set <name> except for the transport type which is SSH instead of telnet as in <name>.

Assigning a Credential Set

The attributes defined in a credential set are used to login to a device, and to perform SNMP operations.

A credential set can be assigned to a single device or multiple devices. If there is no credential set assigned to a device, the default credential set will be used.

Credential sets can also be assigned to a group of devices using the ip-range command.

Note If a credential set is assigned to a device using the manage device command and also using the ip-range, the credential set specified in the manage device command will be used.

The commands used to manage the credential sets are detailed in Table 3-2.

Table 3-2 Commands to Manage Credential Sets

Action	Command
To assign a pre-defined credential set to a device. If no credential set is specified, the default credential set is used.	`[SVR:/server](config)# manage device ip_address/dns-name [credential-set name]`
To remove a device from the managed list.	`[SVR:/server](config)# no manage device ip_address`
To assign a pre-defined credential set to a group of devices, between a specified IP range. If no credential set is specified, the default credential set is used. If no name is specified, default is taken as the name of the list.	`[SVR:/server](config)# ip-range {1-10000} from_ip_address to_ip_address credential-set name`
The auto-manage option allows any discovered devices to be added to the managed list automatically. If no name is specified, default is taken as the name of the list.	`[SVR:/server](config)# ip-range {1-10000} from_ip_address to_ip_address credential-set name [auto-manage]`
To remove the IP range specified by the index parameter.	`[SVR:/server](config)# no ip-range {1-10000}`

Credential Sets in a Non Session Based Device Authentication Environment

If the administrator selects non session based device authentication during installation, Cisco E-DI uses credential sets which are centralized (non session based) device credential stores.

Table 3-3 lists the protocols and credentials used in non session based device authentication mode.

See Device Authentication, page 1-5 for more details.

Table 3-3 Protocols and Credentials Used in Non Session Based Device Authentication Mode

Feature	Primary Transport/Credentials	Secondary Transport/Credentials
Discovering the network	Requires SNMP read access to the devices. Credential used—SNMP read community	
Collecting NE basic inventory	Requires SNMP Read access to the devices. Credential used—SNMP read community	

Table 3-3 Protocols and Credentials Used in Non Session Based Device Authentication Mode (continued)

Feature	Primary Transport/Credentials	Secondary Transport/Credentials
Collecting NE file system information	Retrieved through Telnet/SSH CLI. Credentials used—login, password, enable password. Credential used—SNMP read community	Retrieved through SNMP READ Credential used—SNMP read community
Archiving NE configuration	If SNMP write community is configured on Cisco E-DI, the **copy running <tftp://ediserver>** command is issued through SNMP write operation (CONFIG COPY MIB). The device uploads the configuration through TFTP. Credential used—SNMP write community	A Telnet/SSH connection is opened and the **copy running <tftp://ediserver>** command is issued through CLI. The device uploads the configuration through TFTP. Credentials used—login, password, enable password
Retrieving the content of NE files	If SNMP write community is configured on Cisco E-DI, the **copy <file> <tftp://ediserver>** command is issued through SNMP write operation (CONFIG COPY MIB). The device uploads the configuration through TFTP. Credential used—SNMP write community	A Telnet/SSH connection is opened and the **copy <file> <tftp://ediserver>** command is issued through CLI. The device uploads the configuration through TFTP. Credentials used—login, password, enable password
Running EXEC commands on NEs using exec-cmd, connect exec-mode and XMLPI	A Telnet/SSH connection is opened and command is run through CLI. Credentials used—login, password, enable password	
Configuring NEs through NetCLI/XMLPI	Configuration data is saved to a file on Cisco E-DI. If SNMP write community is configured on Cisco E-DI, the **copy <tftp://ediserver> running** command is issued through SNMP write operation (CONFIG COPY MIB). The device downloads the configuration through TFTP. Credential used—SNMP write community	Configuration data is saved to a file on Cisco E-DI. A Telnet/SSH connection is opened and the **copy <tftp://ediserver> running** command is issued through CLI. The device downloads the configuration through TFTP. Credentials used—login, password, enable password

Credential Sets in a Session Based Device Authentication Environment

Cisco E-DI provides session based device authentication which requires a user to enter a login and password when managing devices. The device authentication login and password are valid for the entire duration of the user session, and are used for authenticating all the devices.

In session based device authentication mode:

- For system initiated tasks or scheduled tasks—All credentials used for Cisco E-DI to device communication are from the central credential set, not from the session credential set.
- For user initiated tasks—SNMP credentials and the enable password are from the central credential set, and the Telnet login and password are from the session.

Table 3-4 lists the protocols and credentials used in session based device authentication mode.

See Device Authentication, page 1-5 for more details.

Table 3-4 Protocols and Credentials Used in Session Based Device Authentication Mode

Feature	Primary Transport/Credentials	Secondary Transport/Credentials
Discovering the network	Requires SNMP read access to the devices. Credential used—SNMP read community	
Collecting NE basic inventory	Requires SNMP Read access to the devices. Credential used—SNMP read community	
Collecting NE file system information	Retrieved through Telnet/SSH CLI if user initiated. Credentials used—login, password, enable password.	
Archiving NE configuration	If the configuration archive is system initiated, and if SNMP write community is configured on Cisco E-DI, the **copy running <tftp://ediserver>** command is issued through SNMP write operation (CONFIG COPY MIB). The device uploads the configuration through TFTP. Credential used—SNMP write community For user initiated configuration archival SNMP write is not used (see secondary credentials).	A Telnet/SSH connection is opened and the **copy running <tftp://ediserver>** command is issued through CLI. The device uploads the configuration through TFTP. Credentials used—login, password, enable password

Table 3-4 Protocols and Credentials Used in Session Based Device Authentication Mode (continued)

Feature	Primary Transport/Credentials	Secondary Transport/Credentials
Retrieving the content of NE files	If the task is system initiated, and if SNMP write community is configured on Cisco E-DI, the **copy <file> <tftp://ediserver>** command is issued through SNMP write operation (CONFIG COPY MIB). The device uploads the configuration through TFTP. Credential used—SNMP write community For user initiated configuration archival SNMP write is not used (see secondary credentials).	A Telnet/SSH connection is opened and the **copy <file> <tftp://ediserver>** command is issued through CLI. The device uploads the configuration through TFTP. Credentials used—login, password, enable password
Running EXEC commands on NEs using exec-cmd, connect exec-mode and XMLPI	A Telnet/SSH connection is opened and command is run through CLI. Credentials used—login, password, enable password	
Configuring NEs through NetCLI/XMLPI	Configuration data is saved to a file on Cisco E-DI. A Telnet/SSH connection is opened and the **copy <tftp://ediserver> running** command is issued through CLI. The device downloads the configuration through TFTP. Credentials used—login, password, enable password	

Comparing Credential Sets in a Non Session Based and Session Based Device Authentication

Table 3-5 details and compares the way that components in a credential set function in non session based device authentication and session based device authentication modes.

See Device Authentication, page 1-5 for more details.

Table 3-5 Comparing the Credentials Used in Non Session Based and Session Based Device Authentication

Type of Credential	Non Session Based Mode	Session Based Mode
SNMP read community	Uses the read community of the credential set configured in the running-config. This can be configured on an individual device basis. Users with at least read-access, can see this credential in an encrypted form in the Cisco E-DI running-config and startup-config.	
SNMP write community	Uses the write community of the credential set configured in the running-config. This can be configured on an individual device basis. Users with at least read access, can see this credential in an encrypted form in the Cisco E-DI running-config and startup-config.	
Transport Type: Telnet or SSH. (This is not a credential)	Uses the transport field of the credential set configured in the running-config. This can be configured on an individual device basis.	
CLI login	Uses the login field of the credential set configured in the running-config. This can be configured on an individual device basis. Users with at least read access, can see this credential in an encrypted form in the Cisco E-DI running-config and startup-config.	Uses the login set in the **terminal device-auth** command by the user in the session. The same login applies to all the devices. The login is stored only in Cisco E-DI memory. It is not visible to any user in any form.
CLI password	Uses the password field of the credential set configured in the running-config. This can be configured on an individual device basis. Users with at least read access, can see this credential in an encrypted form in the Cisco E-DI running-config and startup-config.	Uses the password set in the **terminal device-auth** command by the user in the session. The same password applies to all the devices. The password is stored only in Cisco E-DI memory. It is ot visible to any user in any form.
CLI enable password	Uses the enable-password field of the credential set configured in Cisco E-DI running-config (can be configured on a per-device basis) Users with at least read access, can see this credential in an encrypted form in the Cisco E-DI running-config and startup-config.	

Device Discovery

Basic network discovery is required primarily in situations where Cisco E-DI is deployed without a management application. Discovery is also useful in cases where a non-Cisco management application is deployed in conjunction with Cisco E-DI, and it lacks adequate discovery capabilities. Table 3-6 details the commands required to set up device discovery.

Devices need to be discovered first before they are managed.

Note Discovery can only be triggered from the CLI.

Two mechanisms for discovery are provided:

- Cisco Discovery Protocol (CDP)
- SNMP sweep

Both these mechanisms require that Cisco E-DI have SNMPv1/v2c read access to the NE. Discovered devices are not automatically managed. Devices need to be selected from the discovered list to be managed by Cisco E-DI.

Discovery with a specified frequency can be scheduled.

This section includes the following information:

Setting Up Device Discovery

For a CDP based discovery, a seed IP address has to be provided to start discovering the network. Multiple seed addresses can also be specified to make discovery quicker. A maximum hop count/distance of any discovered device from the seed IP addresses can be specified. The maximum hop count is 10. If no hop count is specified, a default value 1 is used and the hop count is the same for all the seeds specified. Discovery is performed starting from the seed IP addresses specified till all the devices are discovered or the hop count is reached. In server configure mode, discovery can be scheduled with a list of seed IP addresses, hop count and repetition frequency.

If the discovered devices have multiple IP addresses, typically only one of those IP addresses is meant for management. When Cisco E-DI has to choose one of the IP addresses for device identification and management, and the configuration command **discovery use-mgmt-ip-address** is enabled, it uses the following criteria to determine the management interface address:

1. If a loopback IP address (interface) is configured then this is the preferred management IP address.
2. If a device has multiple loopback IP addresses (interfaces), the first address that gets resolved to a hostname is the management IP address.
3. If a loopback IP addresses cannot be resolved then the preferred IP address is the first configured loopback IP address (based on the ifIndex value).
4. If none of the above rules apply, the preferred IP address is the first configured IP address in the device (based on ifIndex value).

Table 3-6 Commands to Setup Device Discovery

Action	Command
To enter the discovery configuration mode.	`[SVR:/server](config)# discovery`
To find the devices preferred management IP address during discovery. This option is disabled by default.	`[SVR:/server](config)# discovery use-mgmt-ip-address`
To specify the seed IP addresses to be used.	`[SVR:/server](conf-disc)# seed ip_address1 {ip_address2, ….}`
To specify a hop count to use. The default value is 1.	`[SVR:/server](conf-disc)# hopcount {number}`
To specify a repetition frequency in either minutes or hours. **Note** The repetition frequency must be set for a discovery job to run.	`[SVR:/server](conf-disc)# repeat frequency {hours number | minutes number}`
To remove the specified seed IP address, or all IP addresses if no IP address specified.	`[SVR:/server](conf-disc)# no seed {ip_address1, ip_address2, ….}`
To remove the specified hop count.	`[SVR:/server](conf-disc)# no hopcount {number}`
To disable repetition.	`[SVR:/server](conf-disc)# no repeat frequency {hours number| minutes number}`

Discovering Devices

Table 3-7 details how to start the discovery process.

Table 3-7 Commands to Start Discovery

Action	Command
To discover all devices with CDP enabled using the CDP mechanism You need to specify single or multiple seed IP addresses and the hop count to be used. The default hop count is 1.	`[SVR:/server]# discover cdp seed_ip_address [seed_ip_address2 ….] [hopcount number]`
To discover all devices using SNMP scan. For an SNMP based discovery, a range of IP addresses is specified. The discovery process begins with the lower address in the range and terminates at the higher address of the range.	`[SVR:/server]# discover snmp-scan ip_address1 ip_address2`

Any discovery, either scheduled using the configure mode or manually run in the exec mode is implemented in the background. Each discovery job is given a unique task id and the status can be checked using the show discovery command.

Displaying and Importing Discovered Devices

Table 3-8 details the commands required to display and import the discovered devices.

Table 3-8 Commands to Show and Import Discovered Devices

Action	Command
To show the discovery history for all discovery jobs and the list of devices discovered	`[SVR:/server]# show discovery history`
To list all the devices that have been discovered so far and their current status.	`[SVR:/server]# show discovery devices-discovered`
To list all the devices that have been discovered for a given discovery job.	`[SVR:/server]# show discovery devices-discovered [task-id]`
To list all devices that have been discovered, with their preferred management IP address that has been determined.	`[SVR:/server]# show discovery devices-discovered mgmt-ip-binding`
To show discovery task history for a specific discovery job.	`[SVR:/server]# show discovery history [task-id]`
To show the discovery task history about the date/time of implementation and number of devices discovered	`[SVR:/server]# show discovery task-history`
To clear discovery history related information.	`[SVR:/server]# clear discovery history`
To clear the discovered devices list.	`[SVR:/server]# clear discovery devices-discovered`
To import all the devices discovered which are currently un-managed, and set them to managed state.	`[SVR:/server]# import devices from-discovered-list all`
To import the devices selectively. All devices with a manageable state are displayed in the discovery history. Select **y** to manage the device or **n** to skip the device. Select **q** to quit.	`[SVR:/server]# import devices from-discovered-list`
To import devices from an XML or CSV seed file.	`[SVR:/server]# import devices from-seed-file filename`
To import all devices.	`[SVR:/server]# import devices from-seed-file all`
To select the management IP address to be used for device discovery. See Setting Up Device Discovery.	`[SVR: /server](config)# discovery use-mgmt-ip-binding`
To show the discovered devices, and the management IP addresses that are identified for those devices.	`[SVR: /server](config)# show discovery devices-discovered mgmt-ip-binding`

Managing Devices

Cisco E-DI will only establish connections to NEs that are in the managed device list. Cisco E-DI will reject sessions directed to any unmanaged device and display the following error, %no such managed device exists.

Once it starts managing the device, Cisco E-DI to NE communication is independent of any management station to Cisco E-DI communication, and Cisco E-DI manages the device until it is asked to stop.

Note You can clear all previous connections, enter the command **clear status connections**.

All the management tasks can be performed through CLI commands. When Cisco E-DI starts managing an NE, it stores the NE identification information and additional inventory information in the system database.

Once device information is located, Cisco E-DI selects a data model from its device package using the following criteria:

- It ensures that the data model's device family matches the target NE's device family.
- For software version, Cisco E-DI tries to find the exact match. If the exact match cannot be found, then it will find the nearest version of the OS knowledge base from the available pool.
- If the NE's OS version is lower than any available OS version then, the lowest available knowledge base version is selected.

You can start managing a device when a credential set has been applied to the device. See Table 3-2.

Grouping

Cisco E-DI provides the option to create groups. This can be used to manage groups of devices conveniently. See Chapter 1, “Cisco E-DI Concepts” for a detailed explanation of groups in Cisco E-DI.

Table 3-9 details the commands used to manage static groups, and Table 3-10 details the commands to manage dynamic groups.

Table 3-9 Commands to Manage Static Groups

Action	Command
To create a static group. **Note** The group name can have no more than 40 characters.	`[SVR:/server] (config)# static-group group-name`
To enter static group configuration mode.	`[SVR:/server](config)# static-group name`
To include a device or a group of devices or any other group static (other than itself), dynamic or system-defined.	`[SVR:/server](conf-static-group)#include {device ip_address | group name}`
To remove the static group.	`[SVR:/server](config)# no static-group name`
To remove a specific device or group.	`[SVR:/server] (conf-static-group)# no include {device ip_address | group name}`

Table 3-10 Commands to Manage Dynamic Groups

Description	Action
To create a dynamic group. **Note** The group name can have no more than 40 characters.	`[SVR:/server] (config)# dynamic-group group-name`
To enter dynamic group configuration mode.	`[SVR:/server] (config)# dynamic-group name`
To specify a rule to be either included or excluded. See Table 1-2 for device capability options.	`[SVR:/server](conf-dynamic-group)# capability (device-capability)* {include | exclude}`
To specify a range of IP addresses to be included into this group.	`[SVR:/server](conf-dynamic-group)# ip-range index from_ip_address to_ip_address`
To specify a devicename to be included into this group	`[SVR:/server](conf-dynamic-group)# devicename contains name-pattern`
To specify a devicetype name to be included into this group	`[SVR:/server](conf-dynamic-group)# devicetype **devicetype-name`
To remove the dynamic group.	`[SVR:/server](config)# no dynamic-group name`
To remove a capability rule. See Table 1-2 for device capability options.	`[SVR:/server](conf-dynamic-group)# no capability device-capability`
To negate the ip-range rule.	`[SVR:/server](conf-dynamic-group)# no ip-range index`
To negate devicename rule	`[SVR:/server](conf-dynamic-group)# no devicename contains name`
To negate devicetype rule	`[SVR:/server](conf-dynamic-group)# no devicetype device-type`

Sample dynamic group configuration in the running config file:

```
dynamic-group Name
    capability cdp-enabled include
    capability edi-server exclude
    ip-range 1 172.16.0.1 172.16.0.15
    devicename contains ap
!
dynamic-group AllRouters
    capability l3-router include
!
dynamic-group AllCisco2600Routers
    devicefamily Cisco2600
!
dynamic-group AllCisco2621Routers
    devicetype Cisco2621
!
static-group SwitchesAndRouters
    include device 172.16.0.1
    include device 172.16.0.5
    include group Switches
    include group AllRouters
!
dynamic-group AllCiscoIOS
    capability os-type-ios include
!
```

Viewing Devices

After the groups are defined, use the commands in Table 3-11 to view the groups and devices that belong to the group.

When a device is managed, basic information like the device name, software version, type, capabilities are stored in the database. This information changes whenever inventory is performed on the device. When the server is reloaded, the information stored in the database is loaded before an inventory is performed on the device.

Table 3-11 Commands to View Devices

Action	Command
To display all the available groups.	`[SVR:/server]# show groups`
To display devices that belong to a specific group.	`[SVR:/server]# show devices [group name]`
To enter the group specified to perform network level operations.	`[SVR:/server]# network [group name]`

Domain Control

Domain control is a mechanism where a user can perform restricted or controlled operations on NEs grouped under one or more domains with associated privilege levels.

A domain group can consist of multiple groups with individual privileges. See Chapter 4, "Managing Security," for more information about user security and roles. Server privileges are mandatory, with the default privilege level being **NoAccess**. A user can be assigned a domain group so that operations are restricted to the devices and privileges set in the domain group. On invoking a task, Cisco E-DI performs the task only on the devices that the user has privileges for. If a device belongs to more than one device group, the matching entry will be evaluated and the appropriate privileges are enforced.

There are two pre-defined domain groups that allow the administrator to easily configure initial user privileges:

- FULL_CONTROL group allows all possible network and server privileges.
- NO_CONTROL domain group allows no actions in any context.

Unless explicitly assigned, a domain group will have no server and network privileges. When a domain group is deleted, the user assigned to that domain group will be assigned to a NO_CONTROL group. The user will be reassigned to the group if it is added again.

Table 3-12 Commands to Manage Domain Groups

Action	Command
To configure a domain group by name.	`[SERVER](config)# domain-group domain-groupname`
To include a device group by index and privilege level. Administrator option can only be obtained by using the FULL_CONTROL domain group.	`[SERVER](conf-domain)# device-group index device-groupname privileges {NetOperator | NoAccess | ReadOnlyUser}`
To assign server privilege level. Administrator privileges can only be obtained with the FULL_CONTROL domain group.	`[SERVER](conf-domain)# server privileges {NoAccess | ReadOnlyUser}`

Table 3-12 Commands to Manage Domain Groups (continued)

Action	Command
To exclude a device group by index and privilege level.	`[SERVER](conf-domain)# **no device-group index** {**device-groupname** | [privileges [NetOperator | NoAccess | ReadOnlyUser] ]}`
To assign a domain group to a user.	`[SERVER](conf)# **user username domain-group** {**domain-groupname** | [FULL_CONTROL | NO_CONTROL] {**password** [ 0 | 7 ] **password**}`

Sample domain group configuration file:

```
dynamic-group BLDG-2
ip-range 1 192.168.3.1 192.168.3.254
!
dynamic-group BLDG-1
ip-range 1 192.168.2.1 192.168.2.254
!
static-group DALLAS
Include device 192.168.2.5
include group CiscoAP1100
!
domain-group LimitedControl
device-group 1 BLDG-2 privileges NoAccess
device-group 2 BLDG-1 privileges ReadOnlyUser
server privileges NoAccess
!
domain-group DALLAS-Admin
device-group 1 DALLAS privileges NetOperator
server privileges ReadOnlyUser
!
user john domain-group LimitedControl password 7 bdMWc9Axpq9HM
user ann domain-group DALLAS-Admin password 7 bdqE0050W3Qaw
```