CHAPTER 10

Configuring Basic Settings

This chapter describes how to configure basic settings on your ASA that are typically required for a functioning configuration. This chapter includes the following sections:

Configuring the Hostname, Domain Name, and Passwords

This section describes how to change the device name and passwords, and includes the following topics:

Changing the Login Password

To change the login password, enter the following command:

Command	Purpose
`{passwd	password} password`	Changes the login password. The login password is used for Telnet and SSH connections. The default login password is "cisco." You can enter **passwd** or **password**. The password is a case-sensitive password of up to 16 alphanumeric and special characters. You can use any character in the password except a question mark or a space. The password is saved in the configuration in encrypted form, so you cannot view the original password after you enter it. Use the **no password** command to restore the password to the default setting.

Changing the Enable Password

To change the enable password, enter the following command:

Command	Purpose

`enable password password`
Example:
`hostname(config)# passwd Pa$$w0rd`
	Changes the enable password, which lets you enter privileged EXEC mode. By default, the enable password is blank.
	The *password* argument is a case-sensitive password of up to 16 alphanumeric and special characters. You can use any character in the password except a question mark or a space.
	This command changes the password for the highest privilege level. If you configure local command authorization, you can set enable passwords for each privilege level from 0 to 15.
	The password is saved in the configuration in encrypted form, so you cannot view the original password after you enter it. Enter the **enable password** command without a password to set the password to the default, which is blank.

Setting the Hostname

To set the hostname, enter the following command:

Command	Purpose

`hostname name`
Example:
`hostname(config)# hostname farscape`
`farscape(config)#`
	Specifies the hostname for the ASA or for a context.
	This name can be up to 63 characters. A hostname must start and end with a letter or digit, and have as interior characters only letters, digits, or a hyphen.
	When you set a hostname for the ASA, that name appears in the command line prompt. If you establish sessions to multiple devices, the hostname helps you keep track of where you enter commands. The default hostname depends on your platform.
	For multiple context mode, the hostname that you set in the system execution space appears in the command line prompt for all contexts. The hostname that you optionally set within a context does not appear in the command line, but can be used by the **banner** command **$(hostname)** token.

Setting the Domain Name

To set the domain name, enter the following command:

Command	Purpose
domain-name *name* **Example:** `hostname(config)# domain-name example.com`	Specifies the domain name for the ASA. The ASA appends the domain name as a suffix to unqualified names. For example, if you set the domain name to "example.com," and specify a syslog server by the unqualified name of "jupiter," then the ASA qualifies the name to "jupiter.example.com." The default domain name is default.domain.invalid. For multiple context mode, you can set the domain name for each context, as well as within the system execution space.

Setting the Date and Time

This section includes the following topics:

- Setting the Time Zone and Daylight Saving Time Date Range, page 10-3
- Setting the Date and Time Using an NTP Server, page 10-4
- Setting the Date and Time Manually, page 10-6

Setting the Time Zone and Daylight Saving Time Date Range

To change the time zone and daylight saving time date range, perform the following steps:

	Command	Purpose
Step 1	**clock timezone** *zone* [-]*hours* [*minutes*] **Example:** `hostname(config)# clock timezone PST -8`	Sets the time zone. By default, the time zone is UTC and the daylight saving time date range is from 2:00 a.m. on the first Sunday in April to 2:00 a.m. on the last Sunday in October. Where *zone* specifies the time zone as a string, for example, **PST** for Pacific Standard Time. The [**-**]*hours* value sets the number of hours of offset from UTC. For example, PST is **-8** hours. The *minutes* value sets the number of minutes of offset from UTC.
Step 2	To change the date range for daylight saving time from the default, enter one of the following commands. The default recurring date range is from 2:00 a.m. on the second Sunday in March to 2:00 a.m. on the first Sunday in November.	

Command	Purpose
clock summer-time *zone* **date** {*day month* | *month day*} *year hh***:***mm* {*day month* | *month day*} *year hh***:***mm* [*offset*] **Example:** `hostname(config)# clock summer-time PDT 1 April 2010 2:00 60`	Sets the start and end dates for daylight saving time as a specific date in a specific year. If you use this command, you need to reset the dates every year. The *zone* value specifies the time zone as a string, for example, **PDT** for Pacific Daylight Time. The *day* value sets the day of the month, from 1 to 31. You can enter the day and month as **April 1** or as **1 April**, for example, depending on your standard date format. The *month* value sets the month as a string. You can enter the day and month as **April 1** or as **1 April**, depending on your standard date format. The *year* value sets the year using four digits, for example, **2004**. The year range is 1993 to 2035. The *hh:mm* value sets the hour and minutes in 24-hour time. The *offset* value sets the number of minutes to change the time for daylight saving time. By default, the value is 60 minutes.
clock summer-time *zone* **recurring** [*week weekday month hh***:***mm week weekday month hh***:***mm*] [*offset*] **Example:** `hostname(config)# clock summer-time PDT recurring first Monday April 2:00 60`	Specifies the start and end dates for daylight saving time, in the form of a day and time of the month, and not a specific date in a year. This command enables you to set a recurring date range that you do not need to change yearly. The *zone* value specifies the time zone as a string, for example, **PDT** for Pacific Daylight Time. The *week* value specifies the week of the month as an integer between 1 and 4 or as the words **first** or **last**. For example, if the day might fall in the partial fifth week, then specify **last**. The *weekday* value specifies the day of the week: **Monday**, **Tuesday**, **Wednesday**, and so on. The *month* value sets the month as a string. The *hh:mm* value sets the hour and minutes in 24-hour time. The *offset* value sets the number of minutes to change the time for daylight savings time. By default, the value is 60 minutes.

Setting the Date and Time Using an NTP Server

To obtain the date and time from an NTP server, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	**ntp authenticate** **Example:** `hostname(config)# ntp authenticate`	Enables authentication with an NTP server.

Step 2	**ntp trusted-key** *key_id* **Example:** hostname(config)# ntp trusted-key 1	Specifies an authentication key ID to be a trusted key, which is required for authentication with an NTP server. The *key_id* argument is a value between 1 and 4294967295. You can enter multiple trusted keys for use with multiple servers.
Step 3	**ntp authentication-key** *key_id* **md5** *key* **Example:** hostname(config)# ntp authentication-key 1 md5 aNiceKey	Sets a key to authenticate with an NTP server. The *key_id* argument is the ID you set in Step 2 using the **ntp trusted-key** command, and the *key* argument is a string up to 32 characters long.
Step 4	**ntp server** *ip_address* [**key** *key_id*] [**source** *interface_name*] [**prefer**] **Example:** hostname(config)# ntp server 10.1.1.1 key 1 prefer	Identifies an NTP server. The *key_id* argument is the ID you set in Step 2 using the **ntp trusted-key** command. The **source** *interface_name* keyword-argument pair identifies the outgoing interface for NTP packets if you do not want to use the default interface in the routing table. Because the system does not include any interfaces in multiple context mode, specify an interface name defined in the admin context. The **prefer** keyword sets this NTP server as the preferred server if multiple servers have similar accuracy. NTP uses an algorithm to determine which server is the most accurate and synchronizes to that one. If servers are of similar accuracy, then the **prefer** keyword specifies which of those servers to use. However, if a server is significantly more accurate than the preferred one, the ASA uses the more accurate one. For example, the ASA uses a server of stratum 2 over a server of stratum 3 that is preferred. You can identify multiple servers; the ASA uses the most accurate server. **Note** In multiple context mode, set the time in the system configuration only.

Setting the Date and Time Manually

To set the date and time manually, perform the following steps:

Detailed Steps

Command	Purpose

`clock set` *`hh`*`:`*`mm`*`:`*`ss`* `{`*`month day`* `|` *`day month`*`}` *`year`*
Example:
`hostname# clock set 20:54:00 april 1 2004`
	Sets the date time manually.
	The *hh***:***mm***:***ss* argument sets the hour, minutes, and seconds in 24-hour time. For example, enter **20:54:00** for 8:54 pm.
	The *day* value sets the day of the month, from 1 to 31. You can enter the day and month as **april 1** or as **1 april**, for example, depending on your standard date format.
	The *month* value sets the month. Depending on your standard date format, you can enter the day and month as **april 1** or as **1 april**.
	The *year* value sets the year using four digits, for example, **2004**. The year range is from 1993 to 2035.
	The default time zone is UTC. If you change the time zone after you enter the **clock set** command using the **clock timezone** command, the time automatically adjusts to the new time zone.
	This command sets the time in the hardware chip, and does not save the time in the configuration file. This time endures reboots. Unlike the other **clock** commands, this command is a privileged EXEC command. To reset the clock, you need to set a new time with the **clock set** command.

Configuring the Master Passphrase

This section describes how to configure the master passphrase and includes the following topics:

- Information About the Master Passphrase, page 10-6
- Licensing Requirements for the Master Passphrase, page 10-7
- Guidelines and Limitations, page 10-7
- Adding or Changing the Master Passphrase, page 10-7
- Disabling the Master Passphrase, page 10-9
- Recovering the Master Passphrase, page 10-10
- Feature History for the Master Passphrase, page 10-11

Information About the Master Passphrase

The master passphrase feature allows you to securely store plain text passwords in encrypted format. The master passphrase provides a key that is used to universally encrypt or mask all passwords, without changing any functionality. Features that implement the master passphrase include the following:

- OSPF

- EIGRP
- VPN load balancing
- VPN (remote access and site-to-site)
- Failover
- AAA servers
- Logging
- Shared licenses

Licensing Requirements for the Master Passphrase

Model	License Requirement
All models	Base License.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single and multiple context mode.

Adding or Changing the Master Passphrase

This section describes how to add or change the master passphrase.

Prerequisites

- If failover is enabled but no failover shared key is set, an error message appears if you change the master passphrase, informing you that you must enter a failover shared key to protect the master passphrase changes from being sent as plain text.
- This procedure will only be accepted in a secure session, for example by console, SSH, or ASDM via HTTPS.

To add or change the master passphrase, perform the following steps:

Detailed Steps

	Command	Purpose

Step 1	**`key config-key password-encryption`**
	`[`*`new_passphrase`*` [`*`old_passphrase`*`]]`
	
	Example:
	`hostname(config)# key config-key`
	`password-encryption`
	`Old key: bumblebee`
	`New key: haverford`
	`Confirm key: haverford`
		Sets the passphrase used for generating the encryption key. The passphrase must be between 8 and 128 characters long. All characters except a back space and double quotes are accepted for the passphrase.
		If you do not enter the new passphrase in the command, you are prompted for it.
		When you want to change the passphrase, you also have to enter the old passphrase.
		See the "Examples" section on page 10-9 for examples of the interactive prompts.
		Note Use the interactive prompts to enter passwords to avoid having the passwords logged in the command history buffer.
		Use the **no key config-key password-encrypt** command with caution, because it changes the encrypted passwords into plain text passwords. You can use the **no** form of this command when downgrading to a software version that does not support password encryption.

Step 2	**`password encryption aes`**
	
	Example:
	`hostname(config)# password encryption aes`
		Enables password encryption. As soon as password encryption is turned on and the master passphrase is available, all the user passwords will be encrypted. The running configuration will show the passwords in the encrypted format.
		If the passphrase is not configured at the time that password encryption is enabled, the command will succeed in anticipation that the passphrase will be available in the future.
		If you later disable password encryption using the **no password encryption aes** command, all existing encrypted passwords are left unchanged, and as long as the master passphrase exists, the encrypted passwords will be decrypted, as required by the application.

Step 3	**`write memory`**
	
	Example:
	`hostname(config)# write memory`
		Saves the runtime value of the master passphrase and the resulting configuration. If you do not enter this command, passwords in startup configuration may still be visible if they were not saved with encryption before.
		In addition, in multiple context mode the master passphrase is changed in the system context configuration. As a result, the passwords in all contexts will be affected. If the **write memory** command is not entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the **write memory all** command in the system context to save all configurations.

Examples

In the following configuration example, no previous key is present:

```
hostname (config)# key config-key password-encryption 12345678
```

In the following configuration example, a key already exists:

```
Hostname (config)# key config-key password-encryption 23456789
Old key: 12345678
hostname (config)#
```

In the following configuration example, you want to key in interactively, but a key already exists. The Old key, New key, and Confirm key prompts will appear on your screen if you enter the **key config-key password-encryption** command and press **Enter** to access interactive mode.

```
hostname (config)# key config-key password-encryption
Old key: 12345678
New key: 23456789
Confirm key: 23456789
```

In the following example, you want to key in interactively, but no key is present. The New key and Confirm key prompts will appear on your screen if you are in interactive mode.

```
hostname (config)# key config-key password-encryption
New key: 12345678
Confirm key: 12345678
```

Disabling the Master Passphrase

Disabling the master passphrase reverts encrypted passwords into plain text passwords. Removing the passphrase might be useful if you downgrade to a previous software version that does not support encrypted passwords.

Prerequisites

- You must know the current master passphrase to disable it. If you do not know the passphrase, see the "Recovering the Master Passphrase" section on page 10-10.
- This procedure will only be accepted in a secure session, that is, by Telnet, SSH, or ASDM via HTTPS.

Detailed Steps

	Command	Purpose

Step 1	**`no key config-key password-encryption`**
	`[`*`old_passphrase`*`]]`
	
	Example:
	`hostname(config)# no key config-key`
	`password-encryption`
	
	`Warning! You have chosen to revert the`
	`encrypted passwords to plain text. This`
	`operation will expose passwords in the`
	`configuration and therefore exercise caution`
	`while viewing, storing, and copying`
	`configuration.`
	
	`Old key: bumblebee`
		Removes the master passphrase.
		If you do not enter the passphrase in the command, you are prompted for it.

Step 2	**`write memory`**
	
	Example:
	`hostname(config)# write memory`
		Saves the run time value of the master passphrase and the resulting configuration. The non-volatile memory containing the passphrase will be erased and overwritten with the 0xFF pattern.
		In multiple mode the master passphrase is changed in the system context configuration. As a result the passwords in all contexts will be affected. If the **write memory** command is not entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the **write memory all** command in the system context to save all configurations.

Recovering the Master Passphrase

You cannot recover the master passphrase.

If the master passphrase is lost or unknown, you can remove it using the **write erase** command followed by the **reload** command. These commands remove the master key and the configuration that includes the encrypted passwords.

Feature History for the Master Passphrase

Table 10-1 lists each feature change and the platform release in which it was implemented.

Table 10-1 Feature History for the Master Passphrase

Feature Name	Platform Releases	Feature Information
Master Passphrase	8.3(1)	This feature was introduced. We introduced the following commands: **key config-key password-encryption**, **password encryption aes**, **clear configure password encryption aes**, **show running-config password encryption aes, show password encryption**.
Password Encryption Visibility	8.4(1)	We modified the **show password encryption** command.

Configuring the DNS Server

Some ASA features require use of a DNS server to access external servers by domain name; for example, the Botnet Traffic Filter feature requires a DNS server to access the dynamic database server and to resolve entries in the static database. Other features, such as the **ping** or **traceroute** command, let you enter a name that you want to ping or traceroute, and the ASA can resolve the name by communicating with a DNS server. Many SSL VPN and certificate commands also support names.

Note The ASA has limited support for using the DNS server, depending on the feature. For example, most commands require you to enter an IP address and can only use a name when you manually configure the **name** command to associate a name with an IP address and enable use of the names using the **names** command.

For information about dynamic DNS, see the "Configuring DDNS" section on page 12-2.

Prerequisites

Make sure that you configure the appropriate routing for any interface on which you enable DNS domain lookup so you can reach the DNS server. See the "Information About Routing" section on page 21-1 for more information about routing.

Detailed Steps

	Command	Purpose
Step 1	**dns domain-lookup** *interface_name* **Example:** `hostname(config)# dns domain-lookup inside`	Enables the ASA to send DNS requests to a DNS server to perform a name lookup for supported commands.

Step 2	**`dns server-group DefaultDNS`**
	Example:
	`hostname(config)# dns server-group DefaultDNS`
		Specifies the DNS server group that the ASA uses for outgoing requests.
		Other DNS server groups can be configured for VPN tunnel groups. See the **tunnel-group** command in the command reference for more information.

Step 3	**`name-server`** *`ip_address`* `[`*`ip_address2`*`] [...] [`*`ip_address6`*`]`
	Example:
	`hostname(config-dns-server-group)# name-server 10.1.1.5 192.168.1.67 209.165.201.6`
		Specifies one or more DNS servers. You can enter all six IP addresses in the same command, separated by spaces, or you can enter each command separately. The ASA tries each DNS server in order until it receives a response.

Monitoring DNS Cache

The ASA provides a local cache of DNS information from external DNS queries that are sent for certain clientless SSL VPN and certificate commands. Each DNS translation request is first looked for in the local cache. If the local cache has the information, the resulting IP address is returned. If the local cache can not resolve the request, a DNS query is sent to the various DNS servers that have been configured. If an external DNS server resolves the request, the resulting IP address is stored in the local cache with its corresponding hostname.

DNS Cache Monitoring Commands

To monitor the DNS cache, enter the following command:

Command	Purpose
`show dns-hosts`	Show the DNS cache, which includes dynamically learned entries from a DNS server as well as manually entered name and IP addresses using the **name** command.

Feature History for DNS Cache

Table 2 lists each feature change and the platform release in which it was implemented.

Table 2 Feature History for DNS Cache

Feature Name	Platform Releases	Feature Information
DNS Cache	7.0(1)	DNS cache stores responses that allow a DNS server to respond more quickly to queries. We introduced the following command: **show dns host**.