# Cisco Secure Firewall ASA HTTP Interface for Automation

**Last Modified:** 2022-06-21

## Cisco Secure Firewall ASA HTTP Interface for Automation

Normally, network administrators interface with ASAs using CLI and ASDM.

There's often a need to manage many firewalls at the same time or automate some management steps.

One way to interface with most network appliances including ASAs is via CLI. An automated tool could Telnet or SSH into a device, authenticate and execute commands, one at a time. This method has a number of drawbacks, however. The tool must maintain the state of the Telnet and SSH connection, and if that connection is broken, the login process has to be repeated. Using CLI, it is only possible to send one command at a time, so administering many firewalls would be time consuming, especially when the firewalls are some latency away from the management station.

An alternative and more efficient method for interfacing with ASAs is HTTP. Using HTTP, an automation tool can execute commands on the ASAs by accessing specifically formatted URLs. With the HTTP interface, there's also the capability to send multiple commands at a time to significantly increase the efficiency of managing remote firewalls.

This document describes some of the common tasks that can be executed using HTTP. This document shows examples of the command line curl utility. However, the same steps can be easily performed with any HTTP library of a programing language such as python.

## Enable HTTP Access

Create a local user, enable authentication for HTTP, and enable the HTTP server. The HTTP server may already be enabled for the management interface for initial ASDM access if you have a default configuration.

**Note** As a fix https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20190501-asa-csrf security advisory, the ASA only accepts Basic HTTP authentication from clients with specific User-Agent headers. The default list of allowed user strings can be seen by executing **show running-config all http** command:

```
http server basic-auth-client ASDM
http server basic-auth-client CSM
http server basic-auth-client REST API Agent
```

The examples in this document will set the User-Agent to **ASDM** and will use Basic authentication. To allow a different User-Agent string, you can add it using the **http server basic-auth-client** command.

**Procedure**

**Step 1** Create a local user.

**username** *username* [**password** *password*] [**privilege** *priv_level*]

You can alternatively use a AAA server for user authentication.

**Example:**

```
ciscoasa(config)# username api password api privilege 15
```

**Step 2** Enable HTTP authentication using the local database.

**aaa authentication http console LOCAL**

You can alternatively use a AAA server for user authentication.

**Example:**

```
ciscoasa(config)# aaa authentication http console LOCAL
```

**Step 3** Enable the HTTP server and allow access from a network.

**http server enable**

**http** *source_IP_address mask source_interface*

**Example:**

```
ciscoasa(config)# http server enable
ciscoasa(config)# http 0.0.0.0 0.0.0.0 inside
```

**Example**

```
username deanwinchester password 67Impala privilege 15
http server enable
http 10.1.1.0 255.255.255.0 management
aaa authentication http console LOCAL
```

# Configuration Management

This section shows how to use the HTTP interface to view and edit the ASA configuration.

## Retrieve the ASA Configuration

You can retrieve the full, running ASA configuration by accessing the following URL on the ASA: **/admin/config**

### *Example*

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/config
!
ASA Version 9.12(2)9
!
hostname asa1
domain-name vik.local
enable password *** pbkdf2
passwd ** encrypted
multicast-routing
names
name 192.168.1.55 asa1.vblan.com
----- SNIP -----
```

## Execute a Single Command

When accessing the ASA via HTTP, the ASA will accept both EXEC and configuration mode commands in the same manner. It is not necessary to enter configuration mode.

Since the commands are specified directly in the URL, they must be URL encoded. For example, any space needs to be replaced with a + sign. There are many online tools to perform this conversion interactively, for example: https://www.urlencoder.org/

### *Example: Show Command*

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/show+version

Cisco Adaptive Security Appliance Software Version 9.12(2)9
Firepower Extensible Operating System Version 2.6(1.152)
Device Manager Version 7.12(1)

Compiled on Mon 30-Sep-19 13:11 PDT by builders
System image file is "disk0:/asa9-12-2-9-smp-k8.bin"
Config file at boot was "startup-config"

asa1 up 19 days 20 hours

Hardware:   ASA5515, 8192 MB RAM, CPU Clarkdale 3058 MHz, 1 CPU (4 cores)
            ASA: 4096 MB RAM, 1 CPU (1 core)
Internal ATA Compact Flash, 8192MB
BIOS Flash MX25L6445E @ 0xffbb0000, 8192KB

Encryption hardware device : Cisco ASA Crypto on-board accelerator (revision 0x1)
                             Boot microcode        : CNPx-MC-BOOT-2.00
                             SSL/IKE microcode     : CNPx-MC-SSL-SB-PLUS-0005
                             IPSec microcode       : CNPx-MC-IPSEC-MAIN-0026
                             Number of accelerators: 1
Baseboard Management Controller (revision 0x1) Firmware Version: 2.4
--- SNIP ---
```

### *Example: Configuration Command*

Note that the command will not output anything unless there's an error or a warning.

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/domain-name+lab.com
```

### *Example: Command Failure*

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/aaa+authentication+http+console+LOCAL
Range already exists.
```

## Execute Multiple Commands

Using a similar URL syntax to a single command, you can execute several consecutive commands. Multiple commands are separated by forward slashes (/).

***Example***

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/object+network+curl-test/host+1.2.3.4

$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/show+run+object+id+curl-test
object network curl-test
 host 1.2.3.4
```

## Execute Commands in Bulk

When many commands need to be submitted at the same time, the ASA can accept them as an HTTP POST to /admin/config.

***Example***

This example shows how to create an ACL saved in a text file.

```
$ cat config.txt
access-list test123 extended permit tcp any any eq www
access-list test123 extended permit tcp any any eq ftp
access-list test123 extended permit tcp any any eq https

$ curl -k -A ASDM https://api:api@172.31.1.5/admin/config --data-binary @config.txt
Cryptochecksum (changed): 5362c464 e7b04911 a0427d83 367676fc
Config OK

$ curl -k -A ASDM https://api:api@172.31.1.5/admin/exec/show+run+access-list+test123
access-list test123 extended permit tcp any any eq www
access-list test123 extended permit tcp any any eq ftp
access-list test123 extended permit tcp any any eq https
```

# Image Management

The ASA HTTP interface can also be used to manage files in flash memory.

## Download Files

All images can be access from the following URL: **/admin/disk0/***filename*

***Example***

In this example, dap.xml is downloaded.

```
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/disk0/dap.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<dapRecordList>
<dapRecord>
<dapName>
<value>Userspoof</value>
</dapName>
<dapViewsRelation>
<value>and</value>
</dapViewsRelation>
<advancedView>
<value>assert(function()
  return (endpoint.registry["WinUserName"].value ~= aaa.cisco.username)
end)()</value>
--- SNIP ---
```

## Upload Files

Files are uploaded by executing an HTTP POST to **/admin/disk0/***filename*. Be aware that there are no prompts to overwrite existing files; the ASA will silently overwrite them.

***Example***

This example shows how to upload the Secure Client to the ASA.

```
$ curl -k -A ASDM
https://api:api@172.31.1.5/admin/disk0/anyconnect-win-4.9.05042-webdeploy-k9.pkg --data-binary
 @anyconnect-win-4.9.05042-webdeploy-k9.pkg
76380273 bytes uploaded
```

# Certificate Management

There are two types of certificates that are installed on the ASA: Identity and CA.

The ASA accepts Identity certificates as PKCS12 files. Because PKCS12 files are binary, you must first encode it into base64. In Linux/Unix-based systems, you can use the openssl utility to perform base64 encoding. In Windows, use certutil with the -encode option.

When installing certificates on the ASA at the CLI, the ASA uses an interactive prompt to accept the certificate. When applying the certificate using the HTTP interface, you must specify the **nointeractive** keyword.

## Install an Identity Certificate

***Example***

The following is an example showing base64 encoding followed by Identity certificate installation.

```
$ openssl base64 -e -in cert.p12 -out cert.pem
$ echo crypto ca import test-tp pkcs12 pkcs12password nointeractive > cert.txt
$ cat cert.pem >>cert.txt
$ echo quit >>cert.txt
$ cat cert.txt
crypto ca import test-tp pkcs12 Lenovo321 pkcs12password nointeractive
MIILmQIBAzCCC18GCSqGSIb3DQEHAaCCC1AEggtMMIILSDCCBf8GCSqGSIb3DQEH
BqCCBfAwggXsAgEAMIIF5QYJKoZIhvcNAQcBMBwGCiqGSIb3DQEMAQYwDgQIMN+G
--- SNIP ---
HuKDMSUwIwYJKoZIhvcNAQkVMRYEFAR7qSZN47HvCRU/82AiUyRwwyojMDEwITAJ
BgUrDgMCGgUABBSIt7Y5piQ1yqlpPOGZWOAUAXMT+gQIV+Oe+uTn7nwCAggA
```

```
quit
$ curl -k -A ASDM https://api:api@172.31.1.5/admin/config --data-binary @cert.txt
Enter the PKCS12 data in base64 representation....
.INFO: Import PKCS12 operation completed successfully
.
Cryptochecksum (changed): 19dd8321 c0cc9f6a 2690799c e4e58e79
Config OK
```

# Install a CA Certificate

### *Example*

```
$ echo crypto ca trustpoint test-tp >cert.txt
$ echo enrollment terminal >>cert.txt
$ echo crypto ca authenticate test-tp nointeractive >>cert.txt
$ cat ca-cert.pem >>cert.txt
$ echo quit >>cert.txt
$ cat cert.txt
crypto ca trustpoint test-tp
enrollment terminal
crypto ca authenticate test-tp nointeractive
-----BEGIN CERTIFICATE-----
MIIFWTCCBEGgAwIBAgITGQAAAESL6p0sIJ47EgAAAAAARDANBgkqhkiG9w0BAQsF
ADA/MRMwEQYKCZImiZPyLGQBGRYDY29tMRUwEwYKCZImiZPyLGQBGRYFdmJsYW4x
--- SNIP ---
XbyWLIGppq++4MFx2JZ3/cBQ26zuu4PKIHKXWnJdOVsAHLrTnYD5jzJlF2q1d1dP
zf9XqoNta0ArpWGs1jFm9fPG/KvgK5iGEmPwbT4=
-----END CERTIFICATE-----
quit

$ curl -k -A ASDM https://api:api@172.31.1.5/admin/config --data-binary @cert.txt
Enter the certificate in base64 representation....
End with the word "quit" on a line by itself.

INFO: Certificate has the following attributes:
Fingerprint:     98bc0332 2e157f21 6abfd738 2598145d

Trustpoint CA certificate accepted.

Cryptochecksum (changed): d426ce12 9be43c52 138896b6 7b954a43
Config OK
```