Firepower System Host Input API Guide

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Version 6.5
September 24, 2019

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2019 Cisco Systems, Inc. All rights reserved.

Understanding Host Input

The Firepower Management Center provides a tool for importing data from other sources on your network to augment the monitored host information.

There are two ways to use the Host Input API to submit network map information, either by running the nmimport tool on the Management Center, or by using a remote client. In either case, you specify the network map details in a text file with comma-separated values (CSV format). Using the Host Input Import Tool, page 2-1 gives general instructions, defines the CSV file format, and describes how to use the nmimport tool. Configuring Host Input Clients, page 3-1 describes how to use a Host Import client.

For example, if you are setting up a new Firepower System, you might want to make sure that all the computers listed in your asset management software exist in the network map. You could export the host data from the asset management application, format the results into an appropriately formatted text file, and import the host data using the host input import tool. If the asset management system includes operating system information for each host, you can set up a third-party product map for the asset management system and map each third-party operating system label to the corresponding Cisco label. You can set that map before you run the import, and the system will associate the appropriate Cisco operating system definition with each host.

There are four major steps to using the host input import tool with the Firepower Management Center:

1. If you want to perform impact correlation using third-party host data, you can configure third-party product maps to map service, operating system, or fix definitions to Cisco product or fix definitions, using the Management Center web interface.
2. If you want to import third-party vulnerabilities, you can configure third-party vulnerability maps to map third-party vulnerability identification strings to Cisco vulnerability IDs, using the Management Center web interface. Note that you can also perform this mapping in your import file.
3. Export data from a third-party application and format it as a CSV file, as described in Using the Host Input Import Tool, page 2-1.
4. Submit the CSV file using either the nmimport tool or a Host Input client.

Prerequisites

To understand the information in this guide, you should be familiar with the features and nomenclature of the Firepower Management Center and the function of its components (in particular, the network map), and with the various related event data the system generates. Information about these functions, together with definitions of unfamiliar or product-specific terms, may be obtained from the *Firepower Management Center Configuration Guide*. Additional information about the data fields documented in this guide may be obtained from the *Configuration Guide* as well.

Product Version Compatibility

The following table describes the product version required for various host input functionality:

Table 1-1 Product Version Compatibility

Functionality	Product Version
Host input functionality	Firepower Management Center version 4.9+
Host input external client functionality	Firepower Management Center version 5.0+
Host input mobile device identification functionality	Firepower Management Center version 5.1+
IPv6 address support	Firepower Management Center version 5.2+
Multiple domain support	Firepower Management Center version 6.0+

Document Conventions

The following table lists the names used in this book to describe the various data field formats employed in host input calls.

Table 1-2 Key Value Data Type Conventions

Data Type	Description
uint	Unsigned integer
uint8	Unsigned 8-bit integer
uint32	Unsigned 32-bit integer
string	Variable length bytes containing character data.

Host Input Scripting Resources

The following describes some of the topics explained in the documentation and where to look for more information.

Table 3 Host Input Resources

To learn more about...	Look in...
the host input import tool	Using the Host Input Import Tool, page 2-1
guidelines for writing an import file to use with the host input import tool	Writing Host Input Import Files, page 2-3
syntax for a specific host input function to include in an import file	Host Input Import Syntax, page 2-6
running the host input import tool	Running a Host Input Import, page 2-29
installing, configuring, and running the host input reference client	Using the Host Input Reference Client, page 3-2

Using the Host Input Import Tool

You can import data to the network map by creating an import file and processing it with the host input import tool.

See the following sections for more information:

- Writing Host Input Import Files, page 2-3
- Host Input Import Syntax, page 2-6
- Running a Host Input Import, page 2-29

Preparing to Run Host Input Imports

Some host import operations depend on product mapping information you supply using the Management Center web interface to map third-party product, fix, and vulnerability names and IDs to definitions in the Cisco database. Depending on the data you plan to import, you may need to perform the configuration steps described in the following sections before you run your import:

- Creating a Third-Party Vulnerability Map, page 2-1
- Creating a Third-Party Product Map, page 2-2

Creating a Third-Party Vulnerability Map

If you want to import data including third-party vulnerabilities and use that data for impact correlation, you must create a third-party vulnerability map set before importing the data. The third-party map set allows the system to translate the third-party vulnerability ID to the corresponding Cisco vulnerability ID. If you do not map a third-party vulnerability before import, the vulnerability does not map to a Cisco vulnerability ID and cannot be used for impact correlation. You can create a map set in two ways: using the Management Center web interface or using the `AddScanResult` command. If you import scan results using this command, be sure to edit the source definition for the input source in your network discovery policy to set the identity source type to Scanner.

Third-party vulnerability mappings can be created at any domain level. Use the `SetMap` command to specify a map name to use for the mappings. The map must be defined at the netmap being used in the CSV file or one of its parents.

For more information on mapping third-party vulnerabilities through the web interface, see the *Firepower Management Center Configuration Guide*. For more information on the `SetMap` and `AddScanresult` commands, see Understanding Import File Format, page 2-3.

Creating a Third-Party Product Map

When you import operating system or server data to a host, you can map third-party product name details to a Cisco product definition. You can create a third-party product map through the Management Center web interface.

The third-party product map set allows the system to translate the third-party vendor, product, and version to the corresponding Cisco definition. When you set a third-party product map containing a server definition or an operating system definition, within the same script you can then just define the display strings for a third-party server or operating system when you add or set it using the API.

If you map third-party fixes to Cisco fix definitions using a third-party product map, set the product map, and then add fixes to hosts using the third-party fix name, the system maps the fixes to the appropriate Cisco fix definitions and deactivates vulnerabilities addressed by the fix.

To map a third-party product to a Cisco product definition:

Access: Admin

1. Select **Policies > Application Detectors**, then click **User Third-Party Mappings**.

   The User Third-Party Mappings page appears.

2. You have two choices:

- To edit an existing map set, click **Edit** next to the map set.
- To create a new map set, click **Create Product Map Set**.

  The Edit Third-Party Product Mappings page appears.

3. Type a name for the mapping set in the **Mapping Set Name** field.
4. Type a description in the **Description** field.
5. You have two choices:

- To map a third-party product, click **Add Product Map**.
- To edit an existing third-party product map, click **Edit** next to the map set.

  The Add Product Map page appears.

6. Type the vendor string used by the third-party product in the **Vendor String** field.
7. Type the product string used by the third-party product in the **Product String** field.
8. Type the version string used by the third-party product in the **Version String** field.
9. In the **Product Mappings** section, select the operating system, product, and versions you want to use for vulnerability mapping from the following lists (if applicable):

- Vendor
- Product
- Major Version
- Minor Version
- Revision Version
- Build

- Patch
- Extension

  For example, if you want a host running a product whose name consists of third-party strings to use the vulnerabilities from Red Hat Linux 9, select **Redhat, Inc.** as the vendor, **Redhat Linux** as the product, and **9** as the version.

10. Click **Save**.

    After you create the third-party product map, you can import data using the `SetOS`, `SetService`, or `AddService` commands. Note the third-party product name details and Cisco product definition before importing data.

To locate third-party and Cisco product details:

Access: Admin

1. Select **Policies > Application Detectors**.

   The Application Detectors page appears.

2. Select **User Third-Party Mappings**.

   The Third-Party Product Mappings page appears.

3. Click the edit icon ( ) for your product map set.

   The Edit Third-Party Product Mappings page appears.

4. Click the edit icon ( ) for your product map.

   The Add Product Map pop-up window appears. Note the **Vendor String**, **Product String**, and **Version String** values.

   For more information on mapping third-party products, see the *Firepower Management Center Configuration Guide*.

Writing Host Input Import Files

This chapter provides details on the syntax to import data using the import commands of the host input import tool. When writing your import file, make sure you follow the instructions provided in the following sections:

Understanding Import File Format

In general terms, the import file is a text file containing one command per line, with comma-separated values (CSV format) specifying the command parameters. You must place a few key commands at the beginning of the file, if required by the operations in the file. Those key commands are described here,

while all the other commands are described later in Host Input Import Syntax, page 2-6.

Caution: The system discards any data in the import file that it cannot interpret. To test your import file before running the import, see Testing Your Import on the Management Center, page 2-29.

Host input import files must begin with the SetDomain (if domains are used), SetSource, and SetMap commands, to provide an application source name and to set up third-party product name mappings for the imported data. For more information, see Understanding Import File Format, page 2-3.

After the SetDomain, SetSource, and SetMap commands, you can add additional command lines to the file. Each command line contains a single command with the parameters needed for that command and ends with a hard return. Note that some fields are required only in that you must supply that information to make sure that the host input succeeds and adds meaningful data to the network map. For example, you can add a fix to the system without providing a fix identification number or fix name that matches an existing Cisco fix definition and without mapping the third-party fix to a Cisco fix.

For more information on syntax for individual commands you can include, see the following sections:

- Host Commands, page 2-6
- Server Commands, page 2-9
- Client Application Commands, page 2-13
- Protocol Commands, page 2-15
- Package Fix Commands, page 2-16
- Host Attribute Commands, page 2-18
- Vulnerabilities Commands, page 2-19
- Setting a Third-Party Product Map, page 2-5

To see an example of a complete import file and explanations of each section of the file, see Example Host Input Import File, page 2-24.

Setting the Domain

If your system has domains defined, then you may need to specify a target domain at the beginning of the import file. If neither the client certificate nor the import file specifies a leaf domain, then running the import will fail with an error message.

- The system builds a separate network map for each leaf domain. In a multidomain deployment, you must specify the leaf domain where you are adding network map data.
- If you will be using the Host Input Client to submit the CSV commands, then you can create separate client certificates for each leaf domain. When using such a certificate, all operations will target the certificate's domain. In this case, there is no reason to use the SetDomain command in the script.
- If you intend to use the nmimport tool on a system with domains, then the import file must begin with the SetDomain command. If no leaf domain is specified by the SetDomain command or by the certificate, the import will immediately fail with an error message without processing any commands.
- The domain name must be fully qualified with space-backslash-space between each domain level, such as `Global \ Accounting` or `Global \ Sales \ East`. The capitalization of the domain name must be identical to how the domain was defined.

To set the domain:

For the first line in your import file, use the following syntax:

`SetDomain, DomainName`
where `SetDomain` is the name of the command and *`DomainName`* is the fully qualified leaf domain to which you want to add the imported data.

Setting the Source Type

At the beginning of your import file, you must identify the source type for the data you plan to import. If you import scan results using this command, be sure to edit the source definition for the input source in your network discovery policy to set the identity source type to Scanner.

To set the source type:

1. Add a line to your import file using the following syntax:

   `SetSourceType, Sourcetype`
   where `SetSourceType` is the name of the command and *`Sourcetype`* is the type of source you want to add or use for the imported data. Valid values are `2` (scanner) or `3` (application).

   If you do not use `SetSourceType`, the default is type is `3` (application).

Setting the Source ID

At the beginning of your import file, you must set the source ID for the data you plan to import.

To set the source application name:

1. Add a line to your import file using the following syntax:

   `SetSource, SourceID`
   where `SetSource` is the name of the command and *`SourceID`* is the identification string you want to display as the source application for the imported data.

   The following is an example of the SetSource command:

   ```
   # Set the current SOURCE_ID and Product Map to "Custom Utility"
   SetSource, Custom Utility
   ```
   To see these commands in context in an example file, see Entire Example File, page 2-28.

Setting a Third-Party Product Map

If you are planning to import third-party operating system, server, or fix definitions, you must create a user third-party product map for the third-party names. You can use this command to set the current third-party map for the current session. You create third-party mappings using the Management Center web interface to set up a reusable map between each third-party vendor, product, and version combination and the corresponding Cisco product definition. If you set a third-party map and then add or set host operating system or server data that includes third-party application names included in the map, the system uses the mappings to map the Cisco product definition, and associated vulnerabilities, to each host where the input occurs.

For instance, you could create a map set called `"Custom Utility"`, in which you define the third-party strings as follows:

- Vendor String - Microsoft

- Product String - Win7

You could select the following Cisco product mapping in the map set:

- Vendor - Microsoft, Corp.
- Product - Windows 7
- Patch - SP3

If you set this product map by calling `SetMap, Custom Utility`, it maps `Microsoft Win7` to the VDB entry for the `Microsoft Windows 7` product.

To set the third-party product map set:

1. Add a line to your import file using the following syntax:

   `SetMap, Third-PartyProductMapName`
   where SetMap is the name of the command and *`Third-PartyProductMapName`* is the name of the third-party product map set you want to use for the import.

   For example, you could put the following line of code following the SetSource command:

   `SetMap, Custom Utility`
   You can also use this command to change to a different third-party product map within an import file.

Host Input Import Syntax

After you set the source ID and product map for your import file, as described in Setting the Source ID, page 2-5, you can add lines to your import file to import the specific data you want to add to your network map using various host input commands. Each import command call must end in a hard return and imports one set of import data. For an example of a complete import file, see Example Host Input Import File, page 2-24.

For more information on specific commands you can use, see the following sections:

Host Commands

You can use the host input API to add and remove hosts in the network map and to set operating system definitions for hosts.

For more information on host commands, see the following sections:

AddHost

You can use the `AddHost` command to add a host to the network map. You can add an IP host (a host with an IP address and optionally a MAC address) or a MAC-only host (a host with only a MAC address). Hosts added in this matter are not subject to the normal host timeout.

The AddHost command will have no impact if the network map already contains a host with the specified IP address or primary MAC address. If your intention is to replace any existing information for a host in the network map with new information, you must use the DeleteHost command before AddHost.

Use this syntax:

```
AddHost, ip_address, mac_address
```

Table 1 AddHost Fields

Field	Description	Required	Values
ip_address	Indicates the IP address for the added host.	Yes (unless a MAC address is provided)	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
mac_address	Indicates the MAC address for the added host.	Yes (unless an IP address is provided)	A single MAC address.

DeleteHost

You can use the `DeleteHost` command to remove a host (or hosts) from the network map. You can remove an IP host (a host with an IP address and optionally a MAC address) by specifying either the IP address or the MAC address for the host. To remove a MAC-only host (a host with only a MAC address), provide the MAC address

Use this syntax:

```
DeleteHost, ip_address, mac_address
```

Table 2 DeleteHost Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes (unless MAC addresses are provided)	A single IP address.
mac_address	Indicates the list of MAC addresses for the affected host or hosts.	Yes (unless IP addresses are provided)	A single MAC address.

SetOS

You can use the `SetOS` command to specify the vendor, product, version, and mobile device information for the operating system for specified hosts. When you import operating system information, you set the display strings for the vendor, product, version, and mobile device information. You can also map the third-party vendor, product, and version strings to a Cisco product definition. See Creating a Third-Party Product Map, page 2-2 for more information.

If you map third-party operating system names to a Cisco definition, the vulnerabilities for that operating system in the Cisco database correspond to the host where the third-party data was imported. If you have already created a third-party product map set using the Management Center web interface, you can use the `SetMap` command to use the values you specified in that map set for the third-party application strings and corresponding Cisco definitions, as described in Setting a Third-Party Product Map, page 2-5.

The operating system identity displayed in a host profile is set by the highest priority source. Possible sources have the following priority order: user, scanner and application (set in the network discovery policy), Firepower, then NetFlow. Note that a new higher priority operating system identity will not override a current operating system identity if it has less detail than the current identity.

If you define a custom operating system for a host, the Management Center web interface indicates the source for the change in the Source Type field of the event view or the basic host information of the host profile.

Use this syntax:

```
SetOS, ip_address, vendor_str, product_str, version_str, vendor_id,
product_id, major, minor, revision, build, patch, extension,
device_string, mobile, jailbroken
```

Or, to set a new product map before you set the operating system, use this syntax:

```
SetMap, map_name
SetOS, ip_address, vendor_str, product_str, version_str, vendor_id,
product_id, major, minor, revision, build, patch, extension,
device_string, mobile, jailbroken
```

For more information on setting third-party product maps, see Setting a Third-Party Product Map, page 2-5.

Table 3 SetOS Fields

Field	Description	Required	Allowed Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
vendor_str	Supplies the operating system vendor display name used by the third-party application.	No	string
product_str	Supplies the operating system product display name used by the third-party application.	No	string
version_str	Supplies the operating system version display name used by the third-party application.	No	string
vendor_id	Supplies the Cisco vendor definition to map to.	No	uint32
product_id	Supplies the Cisco product definition to map to.	No	uint32
major	Supplies the Cisco major version definition to map to.	No	uint32

Table 3 SetOS Fields (continued)

Field	Description	Required	Allowed Values
minor	Supplies the Cisco minor version definition to map to.	No	uint32
revision	Supplies the Cisco revision string to map to.	No	uint32
build	Supplies the Cisco build definition to map to.	No	string
patch	Supplies the Cisco patch definition to map to.	No	string
extension	Supplies the Cisco extension definition to map to.	No	string
device_string	Supplies the detected mobile device hardware information.	No	string
mobile	Indicates whether the operating system is running on a mobile device.	No	uint8
jailbroken	Indicates whether the mobile device operating system is jailbroken.	No	uint8

UnsetOS

You can use the `UnsetOS` command to remove a previously set OS definition from specified hosts. It resets the OS definition to allow the system to track changes to the operating system in the future.

Use this syntax:

```
UnsetOS, ip_address
```

Where *ip_address* is a comma-separated list of IP addresses, CIDR blocks, and ranges of IP addresses representing the host or hosts where you want to reset the operating system identity.

Server Commands

You can update server information for hosts in the network map using the server commands.

For more information, see the following sections:

- AddService, page 2-9
- SetService, page 2-10
- UnsetService, page 2-12
- DeleteService, page 2-12
- Client Application Commands, page 2-13

AddService

You can add a server to an existing host in the network map using the `AddService` command.

The server identity displayed in a host profile is set by the highest priority source. Possible sources have the following priority order: user, scanner and application (set in the network discovery policy), Firepower, then NetFlow. Note that a new higher priority server identity will not override a current operating server identity if it has less detail than the current identity.

Use this syntax:

```
AddService, ip_address, port, proto, server, vendor_str, version_str,
vendor_id, product_id, major, minor, revision, build, patch, extension
```

Or, to set a new product map before you add the server, use this syntax:

```
SetMap, map_name
AddService, ip_address, port, proto, server, vendor_str, version_str,
vendor_id, product_id, major, minor, revision, build, patch, extension
```

For more information on setting third-party product maps, see Creating a Third-Party Product Map, page 2-2 and Setting a Third-Party Product Map, page 2-5.

Table 4 AddService Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	Use this field in combination with the `ip_address` and `proto` fields to specify the server to be added on the hosts where it should be added.	Yes	Integers in the range of 1-65535.
proto	Use this field in combination with the `ip_address` and `port` fields to specify the server to be added on the hosts where it should be added.	Yes	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
server	The name or ID of the server in the Cisco database.	No	To identify the server, you must include a value for either `service_name` or `service_id`. If neither is provided, the server will be listed as `unknown`. If a server name is provided, The system looks up the server ID. If no ID exists for the server name, the system creates an ID.
vendor_str	Supplies the server vendor display name used by the third-party application.	No	string
product_str	Supplies the server product display name used by the third-party application.	No	string
version_str	Supplies the server version display name used by the third-party application.	No	string
vendor_id	Supplies the Cisco vendor definition.	No	uint32
product_id	Supplies the Cisco product definition.	No	uint32
major	Supplies the Cisco major version definition.	No	uint32
minor	Supplies the Cisco minor version definition.	No	uint32
revision	Supplies the Cisco revision string.	No	uint32
build	Supplies the Cisco build definition to map to.	No	string
patch	Supplies the Cisco patch definition to map to.	No	string
extension	Supplies the Cisco extension definition to map to.	No	string

SetService

You can use the `SetService` command to specify the server protocol, vendor, product, and version for a specified server. You can set display strings for the server using service keys. By mapping a third-party product in the Management Center web interface (see Creating a Third-Party Product Map, page 2-2) or using the `SetMap` command (see Setting a Third-Party Product Map, page 2-5), you can associate third-party server data with the vulnerability information for specific Cisco product definitions.

If the server protocol does not already exist, this call causes a new server identity to be created for the string. If the specified server does not exist previously, the system creates it.

The server identity displayed in a host profile is set by the highest priority source. Possible sources have the following priority order: user, scanner and application (set in the network discovery policy), Firepower, then NetFlow. Note that a new higher priority server identity will not override a current server identity if it has less detail than the current identity.

If you define a third-party server definition for a host, the Firepower Management Center web interface indicates the source for the change in the Source Type field of the Servers table view of events or the Servers section of the host profile.

Note: If the number of servers stored in the network map for a specific host exceeds 100, new server information is ignored until servers are deleted from the host.

Use this syntax:

```
SetService, ip_address, port, proto, server, vendor_str, version_str,
vendor_id, product_id, major, minor, revision, build, patch, extension
```

Or, to set a new product map before you set the server, use this syntax:

```
SetMap, map_name
SetService, ip_address, port, proto, server, vendor_str, version_str,
vendor_id, product_id, major, minor, revision, build, patch, extension
```

For more information on setting third-party product maps, see Setting a Third-Party Product Map, page 2-5.

Table 5 SetService Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	Use this field in combination with the `ip_address` and `proto` fields to specify the server to be set on the hosts where it should be set.	Yes	Integers in the range of 1-65535.
proto	Use this field in combination with the `ip_address` and `port` fields to specify the server to be set on the hosts where it should be set.	Yes	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
server	The name or ID of the server in the Cisco database.	No	To identify the server, you must include a value for either `service_name` or `service_id`. If neither is provided, the server will be listed as `unknown`. If a server name is provided, the system looks up the server ID. If no ID exists for the server name, the system creates an ID.
vendor_str	Supplies the server vendor display name used by the third-party application.	No	string
product_str	Supplies the server product display name used by the third-party application.	No	string
version_str	Supplies the server version display name used by the third-party application.	No	string
vendor_id	Supplies the Cisco vendor definition.	No	uint32
product_id	Supplies the Cisco product definition.	No	uint32
major	Supplies the Cisco major version definition.	No	uint32
minor	Supplies the Cisco minor version definition.	No	uint32

Table 5 SetService Fields (continued)

Field	Description	Required	Values
revision	Supplies the Cisco revision string.	No	uint32
build	Supplies the Cisco build definition to map to.	No	string
patch	Supplies the Cisco patch definition to map to.	No	string
extension	Supplies the Cisco extension definition to map to.	No	string

UnsetService

You can use the `UnsetService` command to remove user-added server definitions from a specified host. `UnsetService` does not remove any server definitions detected through Firepower.

Note: If the number of servers stored in the network map for a specific host exceeds 100, new server information is ignored until servers are deleted from the host.

Use this syntax:

```
UnsetService, ip_address, port, proto
```

Table 6 UnsetService Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	Use this field in combination with the `ip_address` and `proto` fields to specify the server to be removed on the hosts where it should be removed.	Yes	Integers in the range of 1-65535.
proto	Use this field in combination with the `ip_address` and `port` fields to specify the server to be removed on the hosts where it should be removed.	Yes	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).

DeleteService

You can use the `DeleteService` command to remove a server from a specified host. You must specify the port and protocol of the server you want to delete.

Use this syntax:

```
DeleteService, ip_address, port, proto
```

Table 7 DeleteService Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	Use this field in combination with the `ip_address` and `proto` fields to specify the server to be deleted on the hosts where it should be deleted.	Yes	Integers in the range of 1-65535.
proto	Use this field in combination with the `ip_address` and `port` fields to specify the server to be deleted on the hosts where it should be deleted.	Yes	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).

Client Application Commands

You can use the client application commands to modify client application data for hosts in the network map.

For more information, see the following sections:

- AddClientApp, page 2-13
- DeleteClientApp, page 2-14
- DeleteClientAppPayload, page 2-14

AddClientApp

You can use the `AddClientApp` command to add client applications to existing hosts in the network map. If the client application name does not already exist in the Cisco database, the system creates a new entry for the client application.

The client application identity displayed in a host profile is set by the highest priority source. Possible sources have the following priority order: user, scanner and application (set in the network discovery policy), Firepower, then NetFlow. Note that a new higher priority client application identity will not be override a current client application identity if it has less detail than the current identity.

Use this syntax:

```
AddClientApp, ip_address, app_name, app_type, version
```

Table 8 AddClientApp Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
app_name	Indicates the client application name.	Yes	A string consisting of alphanumeric characters or spaces. For existing applications, corresponds to ID values in the database. The system looks up the ID to see if it matches an existing client application ID. If it does not, a new ID is created.
app_type	This field is deprecated.	No	A null value.
version	Indicates the application version.	No	A string consisting of alphanumeric characters or spaces.

DeleteClientApp

You can use the `DeleteClientApp` command to remove a client application from the specified host.

Use this syntax:

```
DeleteClientApp, ip_address, app_name, app_type, version
```

Table 9 DeleteClientApp Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
app_name	Indicates the client application name.	Yes	A string consisting of alphanumeric characters or spaces. For existing applications, corresponds to ID values in the database. The system looks up the ID to see if it matches an existing client application ID. If it does not, a new ID is created.
app_type	This field is deprecated.	No	A null value.
version	Indicates the application version.	No	A string consisting of alphanumeric characters or spaces.

DeleteClientAppPayload

You can use the `DeleteClientAppPayload` command to remove a web application from the specified host.

Use this syntax:

```
DeleteClientAppPayload, ip_address, app_name, app_type, version,
payload_type, payload_id
```

Table 10 DeleteClientAppPayload Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
app_name	Indicates the client application name.	Yes	A string consisting of alphanumeric characters or spaces. For existing applications, corresponds to ID values in the database. The system looks up the ID to see if it matches an existing client application ID. If it does not, a new ID is created.
app_type	This field is deprecated.	No	A null value.
version	Indicates the application version.	No	A string consisting of alphanumeric characters or spaces.

Table 10 DeleteClientAppPayload Fields (continued)

Field	Description	Required	Values
payload_type	Indicates the web application category.	Yes	The number `0`. For existing applications, corresponds to ID values in the database. The system looks up the type to see if it matches an existing web application type. If it does not, a new type is created.
payload_id	Indicates the web application name.	Yes	A string consisting of alphanumeric characters or spaces. For existing applications, corresponds to ID values in the database. The system looks up the ID to see if it matches an existing web application ID. If it does not, a new ID is created.

Protocol Commands

You can use the protocol commands to update protocol information for hosts in the network map.

For more information, see the following sections:

- DeleteProtocol, page 2-15
- AddProtocol, page 2-15

DeleteProtocol

You can use the `DeleteProtocol` command to remove a protocol from the specified IP or MAC host.

Use this syntax:

```
DeleteProtocol, ip_address, mac_address, proto, type
```

Table 11 DeleteProtocol Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes (unless MAC addresses are provided)	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
mac_address	Indicates the list of MAC addresses for the affected host or hosts.	Yes (unless IP addresses are provided)	A list of MAC address strings, with or without separating colons.
proto	Indicates the identification string or name of the protocol to be deleted.	Yes	Valid protocol names consisting of alphanumeric characters or spaces. For transport protocols (`"xport"`), protocols listed in the `/etc/protocols` file are acceptable. For network protocols (`"net"`), see Network Protocol Values, page A-1.
type	Indicates the type of protocol to be deleted.	Yes	`"xport"` or `"net"`

AddProtocol

You can use the `AddProtocol` command to add either a network or transport protocol to an existing host in the network map. You can supply either a protocol ID, a transport protocol name that exists in the `/etc/protocols` file on your Management Center or a network protocol name from Network Protocol Values, page A-1.

Note: You cannot add transport protocols to MAC-only hosts.

Use this syntax:

```
AddProtocol, ip_address, mac_address, proto, type
```

Table 12 AddProtocol Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes (unless MAC addresses are provided)	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
mac_address	Indicates the list of MAC addresses for the affected host or hosts.	Yes (unless IP addresses are provided)	A list of MAC address strings, with or without separating colons.
proto	Indicates the identification string or name of the protocol to be added.	Yes	Valid protocol names consisting of alphanumeric characters or spaces. For transport protocols (`"xport"`), protocols listed in the `/etc/protocols` file are acceptable. For network protocols (`"net"`), see Network Protocol Values, page A-1.
type	Indicates the type of protocol to be added.	Yes	`"xport"` or `"net"`

Package Fix Commands

You can use the Package Fix commands to apply or remove fixes for hosts in the network map for the leaf domain where you import.

For more information, see the following sections:

- AddFix, page 2-16
- RemoveFix, page 2-17

AddFix

You can use the `AddFix` command to map a fix to a specified host or server. You can map a fix using a fix ID from the Cisco vulnerability database (VDB), or using a third-party fix that you map to a fix in the VDB using the Management Center web interface.

When you apply a fix to a host or server, the vulnerability mappings for the system are adjusted and the fixed vulnerabilities are marked as Invalid in the web interface and are not used for impact assessment. However, note that if the applied fix is not applicable to the OS or server identity the fix has no effect.

Use the following syntax:

```
AddFix, ip_address, port, proto, fix_id
```

Table 13 AddFix Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	With the `proto` field, identifies the server affected by the fix on the host where the import occurs.	Yes, if the fix applies to a server	Integers in the range of 1-65535.
proto	With the `port` field, identifies the server affected by the fix on the host where the import occurs.	No	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
fix_id	Indicates the identification string for the fix.	Yes	A Cisco fix identification number or a fix name defined in a third-party product map that you use by calling the `SetMap` command before invoking the `AddFix` command. For more information, see Setting a Third-Party Product Map, page 2-5.

RemoveFix

You can use the `RemoveFix` command to remove a fix mapping from the specified host or server. When you remove a fix, vulnerability mappings are updated accordingly.

Use this syntax:

```
RemoveFix, ip_address, port, proto, fix_id
```

Table 14 RemoveFix Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	With the `proto` field, identifies the server affected by the fix on the host where the import occurs.	Yes, if the fix applies to a server	Integers in the range of 1-65535.
proto	With the `port` field, identifies the server affected by the fix on the host where the import occurs.	No	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
fix	Indicates the identification string for the fix.	Yes	A Cisco fix name or a fix name defined in a third-party product map that you use by calling the `SetMap` command before invoking the `AddFix` command. For more information, see Setting a Third-Party Product Map, page 2-5.

Host Attribute Commands

You can use the host input import tool to set attribute values for the network map for the leaf domain where you import. For more information, see the following sections:

AddHostAttribute

You can use the `AddHostAttribute` command to add text or URL attributes. Note that adding a host attribute does not add a value for the attribute. For more information on setting an attribute value, see SetAttributeValue, page 2-18, below.

Use this syntax:

```
AddHostAttribute, attributename, attributetype
```

where *`attributename`* is the name of the attribute (consisting of alphanumeric characters and spaces.) and *`attributetype`* is the type of attribute (`text` or `URL`).

DeleteHostAttribute

You can use the `DeleteHostAttribute` command to delete attributes.

Use this syntax:

```
DeleteHostAttribute, attributename
```

where *`attributename`* is the name of the attribute. (Valid names consist of alphanumeric characters and spaces.)

SetAttributeValue

You can use the `SetAttributeValue` command to set the value of an existing attribute to the specified value for specified hosts. This command can set the value of user-defined host attributes and the `Criticality` attribute. You can use this command to set the host criticality by using `"criticality"` as the attribute id.

Use this syntax:

```
SetAttributeValue, ip_address, attribute, value
```

Table 15 SetAttributeValue Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.

Table 15 SetAttributeValue Fields (continued)

Field	Description	Required	Values
attribute	Indicates the host attribute name.	Yes	Valid attribute names consisting of alphanumeric characters or spaces.
value	Indicates the host attribute value.	Yes	Valid attribute values for the named attribute, consisting of alphanumeric characters or spaces. If a value is passed in for a list attribute, the value must be an existing named value for the list attribute.

DeleteAttributeValue

You can use the `DeleteAttributeValue` command to remove an attribute value for a host.

Use this syntax:

```
DeleteAttributeValue, ip_address, attribute, value
```

Table 16 DeleteAttributeValue Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
id	Indicates the host attribute name.	Yes	Valid attribute names consisting of alphanumeric characters or spaces.
value	Indicates the host attribute value.	Yes	Valid attribute values for the named attribute, consisting of alphanumeric characters or spaces. If a value is passed in for a list attribute, the value must be an existing named value for the list attribute.

Vulnerabilities Commands

You can use the vulnerabilities commands to update the status of vulnerabilities on a host.

For more information, see the following sections:

- SetInvalidVulns, page 2-19
- SetValidVulns, page 2-20

SetInvalidVulns

You can use the `SetInvalidVulns` command to deactivate vulnerabilities on a host or set of hosts. For the command call to be effective, the vulnerability must exist on the host and be set to valid.

Use this syntax:

```
SetInvalidVulns, ip_address, port, proto, type, vuln_id
```

Table 17 SetInvalidVulns Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	With the `proto` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the fix applies to a server	Integers in the range of 1-65535.
proto	With the `port` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the fix applies to a server	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
vuln_id	Indicates the vulnerability ID for the vulnerability.	Yes	Valid Cisco vulnerability IDs, or mapped third-party vulnerability IDs. For third-party vulnerabilities, note that you must map the third-party vulnerability ID and reference the vulnerability map set in the `vuln_type` field. For more information, see Creating a Third-Party Vulnerability Map, page 2-1.

SetValidVulns

You can use the `SetValidVulns` command to activate vulnerabilities on a host or set of hosts. Once you set a vulnerability as Valid for a host, Management Center assigns a red impact to the event if the SID in the event is mapped to the valid vulnerability. For the command call to be effective, the vulnerability must exist on the host and be set to invalid.

Use this syntax:

```
SetValidVulns, ip_address, port, proto, type, vuln_id
```

Table 18 SetValidVulns Fields

Field	Description	Required	Values
ip_address	Indicates the string containing the IP address or addresses for the affected host or hosts.	Yes	An IP address, or a CIDR mask range, or IP-IP range, or a quoted comma-separated list of such values.
port	With the `proto` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the fix applies to a server	Integers in the range of 1-65535.

Table 18 SetValidVulns Fields (continued)

Field	Description	Required	Values
proto	With the `port` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the fix applies to a server	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
vuln_id	Indicates the vulnerability ID for the vulnerability.	Yes	Valid Cisco vulnerability IDs, or mapped third-party vulnerability IDs. For third-party vulnerabilities, note that you must map the third-party vulnerability ID and reference the vulnerability map set in the `vuln_type` field. For more information, see Creating a Third-Party Vulnerability Map, page 2-1.

Scan Result Commands

You can use the host input import tool to add scan results to your Management Center and to flush the added results to the database. When adding a scan result you can map third-party vulnerabilities in the results to CVE or BugTraq vulnerabilities.

See the following sections for more information:

- AddScanResult Command, page 2-21
- ScanFlush Command, page 2-22
- ScanUpdate Command, page 2-23
- DeleteScanResult Command, page 2-23

AddScanResult Command

You can use the `AddScanResult` command to add scan results from a third-party vulnerability scanner and map each vulnerability to a BugTraq or CVE ID. If you import scan results using this command, be sure to edit the source definition for the input source in your network discovery policy to set the identity source type to Scanner.

Use this syntax:

```
AddScanResult, ipaddr, scanner_id, vuln_id, port, protocol, name,
description, cve_ids, bugtraq_ids
```

Note: How results are added depends on whether you use the ScanUpdate or ScanFlush command. For more information, refer to ScanFlush Command, page 2-22 and ScanUpdate Command, page 2-23.

Table 19 AddScanResult Fields

Field	Description	Required	Allowed Values
ipaddr	Indicates the IP address of the scanned host or hosts.	Yes	A single IP address.
scanner_id	Indicates the scanner ID for the scanner that obtained the scan results.	Yes	'scanner_id' where `scanner_id` is a string indicating the name of the scanner that is the source of the vulnerability data you add. To add scan results from a previously used scanner, indicate the specific scanner name listed in system policies on the Management Center where you added the results. Adding results from a new scanner ID adds that scanner to the system policy. New scanners are added as the lowest priority by default. If you want to change the priority of the scanner, you can do so in the system policy. For more information, see the *Firepower Management Center Configuration Guide*.
vuln_id	Indicates the vulnerability ID for the vulnerability.	Yes	Valid Cisco vulnerability IDs, or mapped third-party vulnerability IDs. If this field, port, protocol, bugtraq_ids, and cve_ids are empty, this is a generic scan result.
port	With the `proto` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the vulnerability applies to a server	Integers in the range of 1-65535.
proto	With the `port` field, identifies the server affected by the vulnerability on the host where the import occurs.	Yes, if the vulnerability applies to a server	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).
name	The name of the vulnerability being imported.	No	A string enclosed in single quotes; for example: 'Using NetBIOS to retrieve info from a Windows host'
description	The description of the vulnerability being imported.	No	A string enclosed in single quotes; for example: `'The following 2 NetBIOS names have been gathered...'`
cve_ids	Space-separated list of CVE vulnerability IDs	No	Valid CVE vulnerability IDs; for example, `'cve_ids: CVE2003-0988'`. If this field, port, protocol, vuln_id, and bugtraq_ids are empty, this is a generic scan result.
bugtraq_ids	Space-separated list of BugTraq vulnerability IDs	No	Valid BugTraq vulnerability IDs; for example, `'bugtraq_ids: 9506'`. If this field, port, protocol, vuln_id, and cve_ids are empty, this is a generic scan result.

ScanFlush Command

After you add scan results to a Management Center using `AddScanResult`, you must use either the `ScanUpdate` or `ScanFlush` command to cause the `AddScanResult` commands to run on the Management Center so the scan results upload to the database.

The `ScanFlush` command does not require any arguments, and you can use it at whatever point in the import file that you want to upload data to the database.

If you use the `ScanFlush` command, it removes any existing scan results from the host and only adds the new results.

ScanUpdate Command

After you add scan results to a Management Center using `AddScanResult`, you must use either the `ScanUpdate` or `ScanFlush` command to cause the `AddScanResult` commands to run on the Management Center so the scan results upload to the database.

The `ScanUpdate` command does not require any arguments, and you can use it at whatever point in the import file that you want to upload data to the database.

If you use the `ScanUpdate` command, it does not remove the existing scan results from the host. It merges the new scan results with the existing scan results.

If you use the `ScanUpdate` command with the `DeleteScanResult` command, it deletes the specific results.

Note that a `ScanUpdate` automatically occurs when an import finishes even if it is not explicitly included in the import file, because the client connection closes.

DeleteScanResult Command

You can use the `DeleteScanResult` command with the `ScanUpdate` command to remove specific scan results from a specific host.

If you supply values for the optional parameters, this restricts results to those matching the parameters. If you do not supply values for the optional parameters, it deletes all results on the specified IP address.

Use this syntax:

```
DeleteScanResult, ipaddr, 'scanner_id', vuln_id, port, protocol
```

Table 20 DeleteScanResult Fields

Field	Description	Required	Allowed Values
ipaddr	Indicates the IP address of the scanned host or hosts.	Yes	A single IP address.
scanner_id	Indicates the scanner ID for the scanner that obtained the scan results.	No	'scanner_id' where `scanner_id` is a string indicating the name of the scanner that is the source of the vulnerability data you add. To add scan results from a previously used scanner, indicate the specific scanner name listed in system policies on the Management Center where you added the results. Adding results from a new scanner ID adds that scanner to the system policy. New scanners are added as the lowest priority by default. If you want to change the priority of the scanner, you can do so in the system policy. For more information, see the *Firepower Management Center Configuration Guide*.
vuln_id	Indicates the vulnerability ID for the vulnerability.	No	A valid third-party vulnerability ID.

Table 20 DeleteScanResult Fields (continued)

Field	Description	Required	Allowed Values
port	With the `proto` field, identifies the server affected by the vulnerability on the host where the import occurs.	No	Integers in the range of 1-65535.
proto	With the `port` field, identifies the server affected by the vulnerability on the host where the import occurs.	No	Either the strings `tcp` or `udp` or the appropriate protocol IDs `6` (tcp) or `17` (udp).

Example Host Input Import File

The following sections illustrate how you might construct an import file to import data using the host input import tool.

The following sections, in sequential order, show each portion of the file:

Example: Setting the Source Domain, Source ID and Product Map

The example script starts with calls to set the domain, the name of the source application, and the product map to be used in the import:

```
# Set the current DOMAIN to Global \ Sales \ East
#
SetDomain, Global \ Sales \ East
# Set the current SOURCE_ID and Product Map to "Asset Management App"
SetSource, Asset Management App
SetMap, Asset Management App
```

This source domain provides the domain in which the host information will be added. The source ID value provides an application name for the system to use in host input events resulting from this import. If you viewed a host input event or a host profile for a host modified using this import, the Source Type value would be `Application: Asset Management App`.

Note that the product map called “Asset Management App” referenced by the SetMap command was created using the Management Center web interface.

Because the third-party product map is the Asset Management App map set, the system maps any third-party operating or server names in the commands contained in the import file to Cisco definitions using product maps or fix maps defined in that map set, as illustrated in Example: Setting the Operating System, page 2-25.

Example: Adding a Host

After the file sets the source application name and third-party product map, commands to import data follow. The data will be added to the network map for the leaf domain specified either with the SetDomain command or the certificate. The first import command is the `AddHost` command:

```
# Add an IP host with no Primary MAC
#
AddHost,1.2.3.4
```

Note that the IP address for the added host is `1.2.3.4` and no primary MAC address is set for the host.

Example: Adding a Protocol to the Host

The next command in the import file adds the `ospf` protocol to the `1.2.3.4` host:

```
# Add the ospf protocol to the host
#
AddProtocol, 1.2.3.4,,ospf,xport
```

Note that the protocol type for the protocol is `xport`.

Example: Adding a Server to the Host

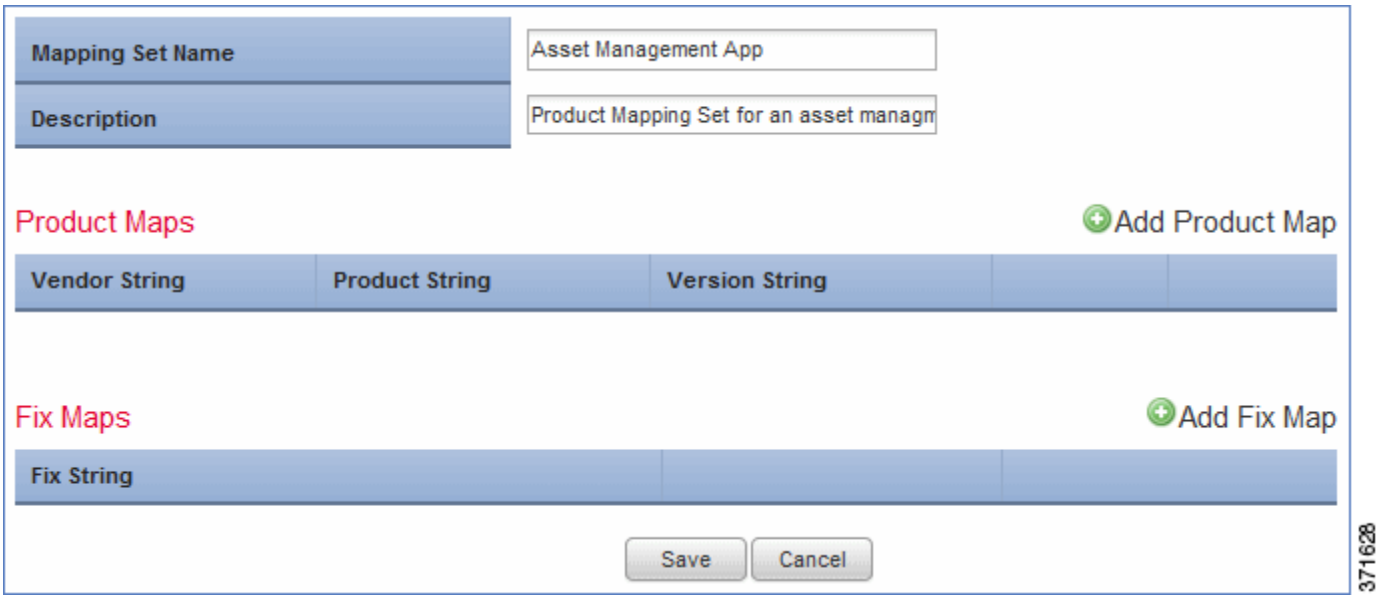
The next command in the import file uses the `AddService` command to add the `OpenSSH` server to the `1.2.3.4` host:

```
# Add a server for the host
#
AddService,1.2.3.4, 22, tcp, ssh, OpenSSH, 4.1
```

Note that the command sets the port to `22`, the protocol to `tcp`, the server type to `ssh`, the vendor display string to `OpenSSH`, and the version display string to `4.1`.

Example: Setting the Operating System

The import file next sets the operating system value for the host using the `SetOS` command. The `Asset Management App` map set contains a product map mapping the third-party product name `Microsoft Win2K` to the Cisco product definition for Microsoft Windows 2000 SP3:



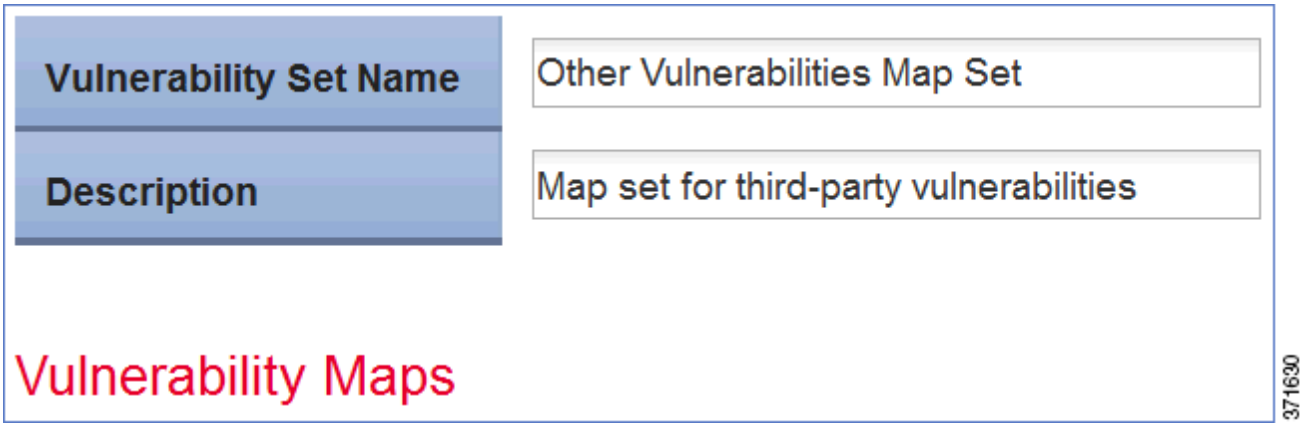
The command in the import file is as follows:

```
# Set the OS. Because the Map is set to "Asset Management App" these values
resolve to the Windows 2000 SP3 definition
#
SetOS, 1.2.3.4, Microsoft, Win2k
```

Note that the `SetOS` command line includes values for the `vendor_str` and `product_str` fields to set the operating system display name to `Microsoft Win2K`. Because those match the **Vendor String** and **Product String** settings defined in the `Asset Management App` product map set, the system maps that third-party operating system name to the Cisco Microsoft Windows 2000 SP3 product definition.

Example: Adding a Third-Party Vulnerability

The import file next imports a third-party vulnerability to the `1.2.3.4` host. This example depends on a third-party vulnerability map set created using the Management Center web interface:



The command in the import file sets the `Vuln003` vulnerability to valid:

```
# Add a third-party vulnerability (from third-party vulnerability map "Other
Vulnerabilities Map Set") to the host
```

```
#
SetValidVuln, 1.2.3.4,,, Other Vulnerabilities Map Set, Vuln0003
```

Example: Setting the Host Criticality

The next command in the import file uses the `SetAttributeValue` command to set the criticality for the `1.2.3.4` host to High.

```
# Set the criticality of the host to "High"
#
SetAttributeValue, 1.2.3.4,criticality,high
```

Note that the attribute name is set to `criticality` and the attribute value is set to `"high"`.

Example: Add Scan Results

The next set of commands in the import file uses the AddHost command to add a host and then the AddScanResult command to add data for that host from a third-party scanner.

```
# Add IP host for scan results to follow
#
AddHost,1.2.3.5
#
# Add the scan result from a Qualys scanner to the network map
#
AddScanResult,1.2.3.5,"Qualys",82003,,,"ICMP Timestamp Request","ICMP
(Internet Control and Error Message Protocol) is a protocol encapsulated in
IP packets. Its principal purpose is to provide a protocol layer able to
inform gateways of the inter-connectivity and accessibility of other
gateways or hosts. ping is a well-known program for determining if a host is
up or down. It uses ICMP echo packets. ICMP timestamp packets are used to
synchronize clocks between hosts.","cve_ids: CVE-1999-0524","bugtraq_ids:"
```

Example: Running Commands on the Management Center

The `ScanFlush` command indicates to the Management Center that it can run the queued commands above the ScanFlush line.

ScanFlush

Example: Adding a Client Application to the Host

The import file then uses the `AddClientApp` command to add a client application named `BMC Remedy` to the `1.2.3.4` host.

```
# Add a Client App
#
AddClientApp, 1.2.3.4, "BMC Remedy", "Asset Manager", "0.0"
```

Note that the client application ID is set to `BMC Remedy`, the client application type is set to `Asset Manager`, and the version is set to `0.0`.

Example: Adding a MAC-Only Host

Finally, the import file uses the `AddHost` command to add a MAC-only host:

```
# Add a MAC-only host
#
```

```
AddHost,,01:02:03:04:05:06
```

Note that the `ip_address` field is left blank and the MAC address is provided instead.

In addition, note that although there is no `ScanFlush` command at the end of the file, the remaining data from the script is sent to the network map when the import file finishes because the session disconnects.

Entire Example File

The full import file explained in the sections above looks like this:

```
# Example import file for Host Input Import Tool
#
# Set the DOMAIN to “Global \ Sales \ East”
#
# Set the current SOURCE_ID and Product Map to "Asset Management App"
#
SetDomain, Global \ Sales \ East
SetSource, Asset Management App
SetMap, Asset Management App
#
# Add an IP host with no Primary MAC
#
AddHost,1.2.3.4
#
# Add the ospf protocol to the host
#
AddProtocol, 1.2.3.4,,ospf,xport
#
# Add a server for the host
#
AddService,1.2.3.4, 22, tcp, ssh, OpenSSH, 4.1
#
# Set the OS. Because the Map is set to "Asset Management App" these values
resolve to the Windows 2000 SP3 definition
#
SetOS, 1.2.3.4, Microsoft, Win2k
#
# Add a third-party vulnerability (from third-party map "Other
Vulnerabilities Set") to the host
#
SetValidVuln, 1.2.3.4,,, Other Vulnerabilities Set, Vuln0003
#
# Set the criticality of the host to "High"
#
SetAttributeValue, 1.2.3.4,criticality,high
#
# Add IP host for scan results to follow
#
AddHost,1.2.3.5
#
# Add the scan result from a Qualys scanner to the network map
#
AddScanResult,1.2.3.5,"Qualys",82003,,,"ICMP Timestamp Request","ICMP
(Internet Control and Error Message Protocol) is a protocol encapsulated in
IP packets. Its principal purpose is to provide a protocol layer able to
inform gateways of the inter-connectivity and accessibility of other
```

```
gateways or hosts. ping is a well-known program for determining if a host is
up or down. It uses ICMP echo packets. ICMP timestamp packets are used to
synchronize clocks between hosts.","cve_ids: CVE-1999-0524","bugtraq_ids:"
#
#Send the commands above to the host input service for processing
#
ScanFlush
#
# Add a Client App
#
AddClientApp, 1.2.3.4, "BMC Remedy", "Asset Manager", "0.0"
#
# Add a MAC only host
#
AddHost,,01:02:03:04:05:06
```

Testing Your Import on the Management Center

You can simulate an import with your import file to make sure it behaves as expected. Because many commands allow you to import duplicate data into the network map, you want to avoid running the same import multiple times. Running a test import avoids that problem. Additionally, the system discards any data in the import file that it cannot interpret, so you want to make sure that the import file will import completely. The test reports the results to the screen (or you can redirect them to a file) so you can then correct any problems with the file before you run the actual import.

To test an import file:

1. Copy the import file you created to the Management Center where you want to run the import.
2. Log into your Management Center with the `admin` account.
3. At the command line, type `nmimport.pl -t` *`filename`*.

   To redirect the results of the test import to a log file, add `>` *`logfilename`* to the end of the command.

   The system adds the imported data to the network map and either displays the result messages on the screen or redirects them to the file you specify.

Running a Host Input Import

You can run the host input import tool from the command line to process the import file you created.

Caution: The system discards any data in the import file that it cannot interpret. Additionally, if you run the same import multiple times, you may find duplicate data in your network map for some items. To prevent these issues, you may want to test import of your import file before running the actual import. For more information, see Testing Your Import on the Management Center, page 2-29.

Note that if you set up the host input reference client on a remote host with access to the Management Center, you can use the `sf_host_input_agent.pl` script to process an import file from the client. For more information on setting up the reference client, see Running the Host Input Reference Client, page 3-4.

To run an import:

1. Copy the import file you created to the Management Center where you want to run the import.
2. Log into your Management Center with the `root` account.

3. At the command line, type `nmimport.pl` *`filename`*.

   To redirect the results of the test import to a log file, add `>` *`logfilename`* to the end of the command.

   The system adds the imported data to the network map and either displays the result messages on the screen or redirects them to the file you specify.

Configuring Host Input Clients

In addition to accepting host input commands from users on the Management Center, the Management Center's host input service also accepts batch import files from authenticated host input clients on external hosts. You can use a host input client to process import files created for the host input import tool and then send the data to the Management Center to add the information to your network map.

You can use the provided host input API reference client to process and send CSV data or to test your host input client connection to the Management Center.

Perform the following tasks to manage Management Center and input client interaction:

1. Establish an authenticated connection to the Management Center.

   See Registering the Host Input Client with the Management Center, page 3-1 for information about generating authentication credentials to establish an authenticated connection to the Management Center.

2. Set up the reference client on the computer where you plan to run it. For more information, see Using the Host Input Reference Client, page 3-2.

   For information on creating import files (also referred to as command files) that you will use your reference client to process, see Writing Host Input Import Files, page 2-3.

Registering the Host Input Client with the Management Center

License: Any

Before you can use a host input client, you must register the computer on which the client runs with the Management Center. The Management Center then generates an authentication certificate, which you download to your client computer.

To add a host input client:

Access: Admin

1. If domains have been created on your system, select the desired domain in the domain switcher. Clients using a certificate created for the Global domain or another parent domain will have permission to alter any leaf domain in that scope, but the import file must specify which domain. Clients using a certificate created for a leaf domain will only have permission to alter that leaf domain.

2. Select **System > Integration > Host Input Client**.

   The Host Input Client page appears.

3. Click **Create Client**.

   The Create Client page appears.

4. In the **Hostname** field, enter the host name or IP address of the host running the host input client.

Note: If you use a host name, the host input server **must** be able to resolve the host to an IP address. If you have not configured DNS resolution, you should configure it first or use an IP address.

5. If you want to encrypt the certificate file, enter a password in the **Password** field.
6. Click **Save**.

   The host input service allows the client computer to access port 8307 on the Management Center and creates an authentication certificate to use during client-server authentication. The Host Input Client page re-appears, with the new client listed under **Host Input Clients**.
7. Click the download icon ( ) next to the certificate file.
8. Save the certificate file to the directory used by your client computer for SSL authentication.

   The client can now connect to the Management Center.

Note: To revoke access for a client, click the delete icon ( ) next to the host you want to remove. Note that you do not need to restart the host input service on the Management Center; access is revoked immediately.

Connecting the Client to the Management Center

The host input service on the Management Center reads a version from the client when the client connects. If the client sends a version newer than the version of the server, the service rejects the connection.

In addition, during the initial exchange, the host input service communicates the maximum allowed data size per transaction to the client. If the client attempts to send a data block bigger than the maximum size, the server closes the connection.

Using the Host Input Reference Client

The reference client provided with the host input SDK is a set of sample client scripts and Perl modules that illustrate how you can use the host input API. You can run them to familiarize yourself with host input import, or you can use them to debug problems with installations of your custom-built client. You can also use one of the scripts to process a host input command file from the client.

For more information on setting up reference clients, see the following sections:

- Setting Up the Host Input Reference Client, page 3-2
- Running the Host Input Reference Client, page 3-4

Setting Up the Host Input Reference Client

To use the host input reference client, you must first install the sample scripts and configure your client to fit the script requirements.

For more information, see the following sections:

- Understanding the Host Input Reference Client, page 3-3
- Configuring Communications for the Host Input Reference Client, page 3-3

Understanding the Host Input Reference Client

You can download the `HostInputClientSDK.zip` package, which contains the host input reference client, from the Cisco support site. The Table 3-1Host Input Reference Client Files, page 3-3 lists the files included in the `HostInputClientSDK.zip` package.

Table 3-1 Host Input Reference Client Files

Filename	Description
SFHIClient.pm	This Perl module contains the commands called by the Perl clients.
SFPkcs12.pm	This Perl module parses the client certificate and allows the client to connect to the Management Center.
sf_host_input_agent.pl	You can use this Perl script to import CSV data by specifying the appropriate input plugin and a command file.
InputPlugins/csv.pm	You can call this Perl module to run a command file that imports CSV data.

Configuring Communications for the Host Input Reference Client

The reference client uses the Transport Layer Security (TLS) 1.2 protocol for data communication. You must install OpenSSL on the computer you plan to use as a client and configure it appropriately for your environment.

To set up SSL on your client:

1. Download OpenSSL from `http://openssl.org/source/`.
2. Unpack the source to `/usr/local/src`.
3. Configure the source by running the Configure script.
4. Make and install the compiled source.

Loading General Prerequisites for the Host Input Reference Client

Before you can run the host input reference client, you must install the `IO::Socket::SSL` Perl module on the client computer. You can install the module manually or use `cpan` to do so.

Note: If the `Net::SSLeay` module is not installed on the client computer, install that module as well. `Net::SSLeay` is required for communication with OpenSSL.

You also need to install and configure OpenSSL to support an SSL connection to the Management Center. For more information, see Configuring Communications for the Host Input Reference Client, page 3-3.

In addition, if you plan to use the Qualys plugin with the host input client, you must install the `XML::Smart` Perl module and its prerequisites. If you plan to use IPv6 to communicate between the client and the Management Center, you must also install the `IO::Socket::INET6` Perl module.

Downloading and Unpacking the Host Input Reference Client

You can download the `HostInputClientSDK.zip` file that contains the host input reference client from the Support site.

Unpack the zip file to a computer running the Linux operating system, where you plan to run the client.

Creating a Certificate for the Host Input Reference Client

License: Any

Before you can use the host input reference client, you need to create client certificate as described in Registering the Host Input Client with the Management Center, page 3-1. You must save that certificate file to the directory where you put the reference client.

To create a certificate for the reference client:

Access: Admin

1. Create a client as described in Registering the Host Input Client with the Management Center, page 3-1.
2. Save the certificate file to the directory where you put the reference client.

Running the Host Input Reference Client

The Host Input Perl reference client scripts are designed for use on an operating system with the Linux kernel but should work on any POSIX-based operating system, as long as the client machine meets the prerequisites defined in Setting Up the Host Input Reference Client, page 3-2.

You can use the reference client to import CSV data from a remote client to the network map on a Management Center.

Use the following syntax to run the `sf_host_input_agent.pl` script:

```
./sf_host_input_agent.pl -server=ManagementCenterIPAddress -level=DebugLevel
-logfile=LogFile -pl CSVCommandFile.csv csv
```

For example, to import using a CSV file named `csv_file.txt` to a Management Center with an IP address of 10.10.0.4 and with debug logging to HostInput.log log file:

```
./sf_host_input_agent.pl -server=10.10.0.4 -level=3 -logfile=HostInput.log
-pl -pluginfo=cvs_file.txt csv
```

Network Protocol Values

Using the `AddProtocol` and `DeleteProtocol` commands, you can add protocols to or remove protocols from hosts. The following table details the available network protocol values.

Table A-1 Network Protocol Values

Value	Description
IP	Internet Protocol version 4
ARP	Address Resolution Protocol
BPDU(STP)	Bridge Protocol Data Unit (Spanning Tree Protocol)
RARP	Reverse Address Resolution Protocol
OldIPX	Internetwork Packet Exchange, older version
IP Version 6	Internet Protocol version 6
Loopback	Loopback
SNAP	Subnetwork Access Protocol
Novell NetWare	Novell NetWare
NetBIOS	Network Basic Input/Output System
NetBIOS (Response)	Network Basic Input/Output System response
IPX	Internetwork Packet Exchange
Intel ANS	Intel Advanced Network Services
DEC MOP Dump/Load Assistance	Digital Equipment Corporation Maintenance Operations Protocol dump/load assistance
DEC MOP Remote Console	Digital Equipment Corporation Maintenance Operations Protocol remote console
PPPoE Discovery	Point-to-Point over Ethernet discovery stage
PPPoE Session	Point-to-Point over Ethernet session stage