ForeScout

Network Function Property Algorithm

CounterACT® Technical Note

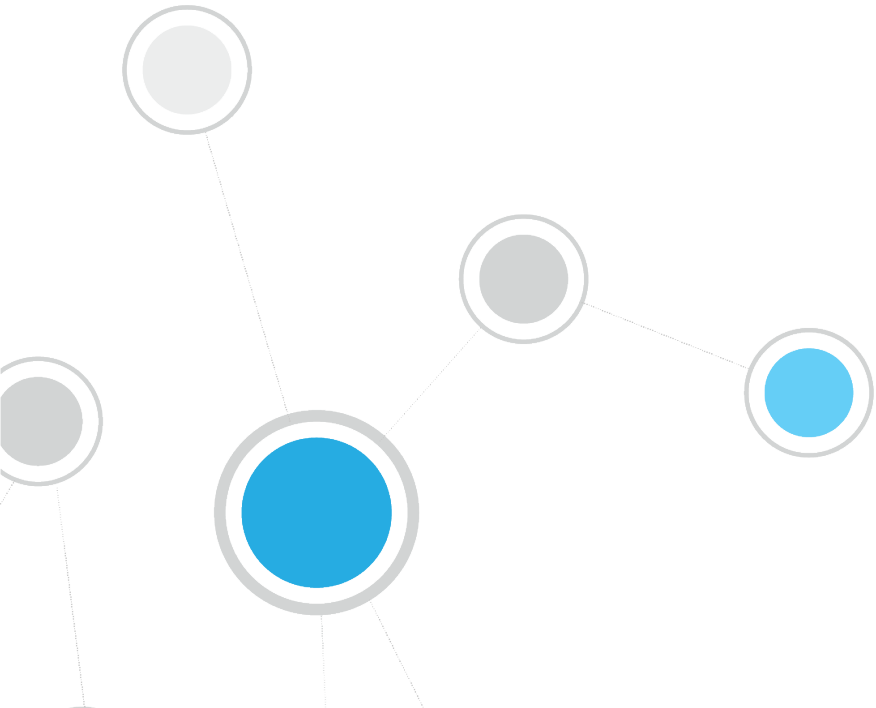
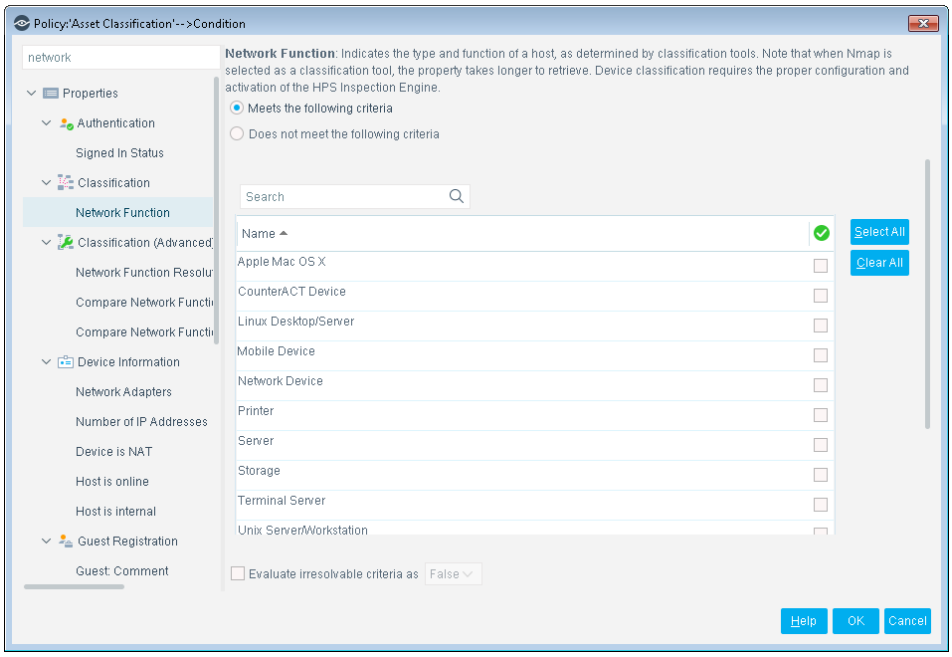


Table of Contents

About the Network Function Property

The Network Function host property is used in ForeScout CounterACT® policies to detect endpoint network function values, for example Windows machines, mobile devices or network printers. This property can be used to help you continuously track and control your network assets.



The following property values may be resolved:

- Windows Machine
- Unix Server/Workstation
- Server
- Printer
- Mobile Device
- VoIP Device
- Linux Desktop/Server
- Apple Mac OS X
- Terminal Server
- Storage
- CounterACT Device
- Network Devices, switches, routers and storage devices

If a device does not meet the criteria for any of the above values it is resolved as *irresolvable*.

Network Function Algorithm Criteria

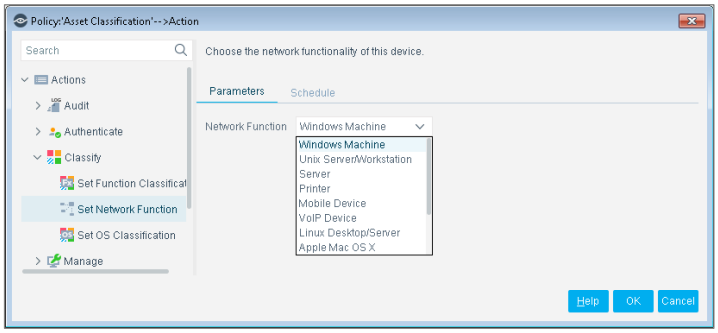
Network Function property resolution is based on a hierarchal algorithm. The algorithm applies a series of inspection criteria on each endpoint. If the endpoint does not match the first criterion, the next possible criterion is inspected.

The Network Function is resolved when the first matching item in the list of criteria matches a value discovered on the endpoint. For example, manually classifying an endpoint as a *Windows Machine* will override any property resolution determined using the algorithm.

This section lists the hierarchy of the criteria applied and describes each criterion.

1. Manual Classification

Derived from the *Classify* > *Set Network Function* action in the CounterACT Console.



2. Managed CounterACT Appliance

The Enterprise Manager resolves itself as a managed CounterACT Appliance, as well as any Appliances it manages. Standalone Appliances resolve themselves as managed CounterACT Appliances as well.

3. Managed Endpoint

Derived from CounterACT managing an endpoint.

- **a.** Managed by HPS using RI (Windows Machine)
- **b.** Managed by HPS using SC (Windows Machine)
- **c.** Managed by HPS Windows CE Agent (Mobile Device)
- **d.** Managed by Mac/Linux Plugin Remote Inspection SSH (Linux Desktop/Server | Unix Server/Workstation | Apple Mac OS X)
- **e.** Managed by Mac/Linux Plugin SecureConnector (Linux Desktop/Server | Unix Server/Workstation | Apple Mac OS X)
- **f.** Managed by Android Plugin SecureConnector (Mobile Device)
- **g.** Managed by Fiberlink Plugin (Mobile Device)
- **h.** Managed by iOS Plugin (Mobile Device)
- **i.** Managed by Other MDM Plugin (Mobile Device)

4. Switch Plugin

Any devices managed by the Switch Plugin are resolved as *Network Device.*

5. VPN Plugin

Any devices managed by the VPN Plugin are resolved as *Network Device*.

6. Wireless Plugin

Any devices managed by the Wireless Plugin are resolved as a *Network Device.*

7. Switch CDP

The Switch Plugin may report information about devices learned via CDP from managed switches. This information can be used to determine the network function of those devices.

8. VoIP Connected to Managed Switch

A device managed by the Switch Plugin determines that an attached endpoint is resolved as a *VoIP Device*.

9. Trusted Vendor MAC

CounterACT determines the NIC vendor based on the IEEE list of MAC address prefixes. CounterACT has a list of reliable vendor name matches which it will match against. Currently, this list matches against certain MAC ranges assigned to HTC and RIM only, and resolves such endpoints as *Mobile Device*.

In addition to the predefined matching performed by CounterACT, in cases where other network devices can be resolved based on a known NIC vendor or MAC prefix, this raw data can be used in custom policies to classify devices.

10. DHCP Classification

Resolved by using passive fingerprinting of DHCP traffic. This relies on the OS type calculated by the DHCP Classifier Plugin based on the DHCP traffic it sees.

In addition to the predefined matching performed by CounterACT, in cases where other network devices can be resolved based on DHCP traffic information, the raw data (such as options and request fingerprints) can be used in custom policies to classify devices.

11. Passive Detection of HTTP User Agent

The Packet Engine listens to HTTP traffic and notes the user agent header in HTTP requests. Based on defined translation rules, the user agent is used to resolve the value of the network function.

In addition to the predefined matching performed by CounterACT, in cases where other network devices can be resolved based on string matching within the HTTP user agent, this raw data can be used in custom policies to classify devices.

12. Samba Traffic

The Packet Engine passively listens to Samba traffic and based on unique signatures in the traffic, determines the value of the network function. This is useful for determining *Windows Machine* devices.

13. HTTP User Agent on CounterACT Web Server

Same as 11. Passive Detection of HTTP User Agent, except that instead of passively listening for traffic, the HTTP User Agent of the client connecting to the CounterACT web server is read (eg., during a HTTP hijack or compliance portal access). The same rules are then used to translate this into a defined network function.

In addition to the predefined matching performed by CounterACT, in cases where other network devices can be resolved based on string matching within the HTTP user agent, this raw data can be used in custom policies to classify devices.

14. Passive Fingerprinting

The Packet Engine performs passive fingerprinting of the SYN and SYN-ACK packets.

15. Nmap Active Banner Scan

Resolved by CounterACT when it actively scans the service banners of an endpoint using Nmap.

In addition to the predefined matching performed by CounterACT, in cases where other network devices can be resolved based on string matching to TCP service banners, this raw data can be used in custom policies to classify devices.

16. Nmap Active OS Scan

Resolved by CounterACT when it actively scans the OS fingerprint of an endpoint using Nmap. This is done by analyzing the fingerprint of the TCP/IP stack.

Property Resolution Method

The lookup table typically has a mapping between regular expressions and network function values. If a given input does not match any regular expression in the lookup table, that item is considered to not have resolved the network function and the following item, in order of priority, is used.

Example 1

The classification at criterion 9. Trusted Vendor MAC, takes the vendor name of the NIC (based on the IEEE MAC address lookup table) and then considers whether it matches the following regular expression:

^htc corporation$|^rim$|^RESEARCH IN MOTION$|^RESEARCH IN MOTION LIMITED$

If it does, then the resulting value is 'Mobile Device'.

Example 2

The inspection at criterion 11. Passive Detection of HTTP User Agent and 13. HTTP User Agent on CounterACT Web Server, where the User Agent string of the HTTP request is matched against the following regular expressions to get corresponding property values:

*Regular Expression	Resulting Value
Windows\s*(Mobile|Phone)	Mobile Device
/.*bsalsa.*|.*kerberos-sec.*//Microsoft|Windows	Windows Machine
Jetdirect	Printer
Netware	Server
VoIP|IP\s*Phone	VoIP Device
(^|\W)hp.*switch>hp~netdevice,iPhone|iPad|iPod|Mobile.*Darwin	Mobile Device
Android|Maemo>handheld~linux,Black\s*berry|Symbian|HP.*Palm | Samsung.*phone|Cisco\s*Cius|Amazon\s*Kindle|MIDP|Hand\s*Held	Mobile Device
/(tcp\s*|Openwall GNU.*)Linux telnetd|Apache httpd [\s\d.()]*Unix.*OpenSSL//RHN-	Linux Desktop/Server
(^|\W)emc($|\W)	Storage
juniper|aruba|meru|router|wap|switch|wireless|bridge| Integrated\s*Lights\W*Out|Allegro(Soft)?\s*Rom|ConnectUPS|VxWork	Network Device

* This is not a complete list.

Legal Notice

2018-04-10 09:21