Cisco Plug and Play Feature Guide

Cisco Services

Cisco Plug and Play Feature Guide

TABLE OF CONTENTS

Contents

Cisco Plug and Play Feature Guide

INTRODUCTION

Introduction

The Cisco® Plug and Play solution is a converged solution that provides a highly secure, scalable, seamless, and unified zero-touch deployment experience.

Enterprises incur major operating costs to install and deploy networking devices as part of campus and branch deployments. Typically, every device has to be pre-staged, which involves repetitively copying Cisco IOS® Software images and applying configurations manually through a console connection. Once pre-staged, these devices are then shipped to the final site for installation. The end-site installation may require a skilled installer for troubleshooting, bootstrapping, or modifying the configuration. The entire process can be costly, time-consuming, and prone to errors. At the same time, customers would like to increase the speed and reduce complexity of the deployment without compromising security.

Cisco Plug and Play Components

The Cisco Plug and Play (PnP) deployment includes a PnP agent, a PnP server, and other components.

This simplified deployment process automates the following deployment-related operational steps, on Cisco devices:

- Establishing initial network connectivity for the device
- Delivering device configuration
- Delivering software and firmware images
- Delivering licenses
- Delivering deployment script files
- Provisioning local credentials
- Notifying other management systems about deployment related events
-

Plug-n-Play Agent

The Cisco Plug and Play (PnP) agent is an embedded software component that is present in all Cisco network devices that support simplified deployment architecture. The PnP agent understands and interacts only with a PnP server. The PnP agent, using DHCP, DNS or other such methods, tries to acquire the IP address of the PnP server with which it wants to communicate. After a server is found and a connection has been established, the agent communicates with the PnP server to perform deployment-related activities.

It also notifies the server of all interesting deployment-related events like out-of-band configuration changes and a new device connection on an interface.

Key Benefits

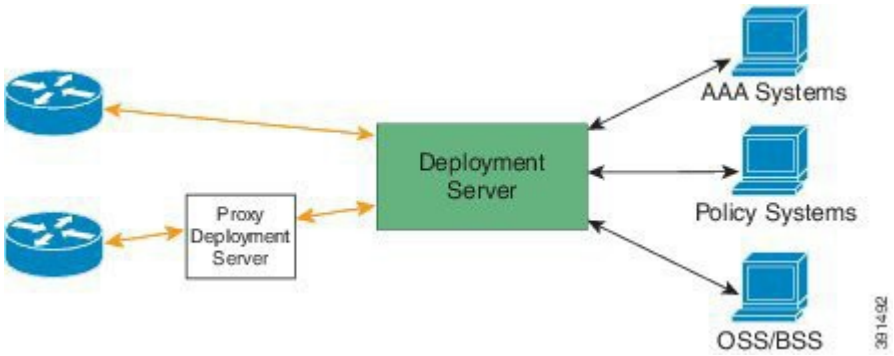
The Cisco Plug and Play (PnP) agent provides you the following benefits:

- Day 0 bootstrapping—Configuration, image, licenses, and other files
- Day 2 management—Configuration and image upgrades and on-going monitoring of Simple Network Management Protocol (SNMP) and syslog messages
- Open communication protocol—Enables customers and partners to write applications
- XML based payload over HTTP and Extensible Messaging and Presence Protocol (XMPP) between the server and the agent
- Security—Authentication and an encrypted communication channel between the management app and the agent
- Deployment and management of devices behind the firewall and Network Address Translation (NAT)
- Support for one-to-one and one-to-many communication
- Support for policy based deployment (product ID or location of the device)
- Deployment based on unique ID (Unique Device Identifier [UDI] or MAC)
- Unified solution across Cisco platforms (including IOS classic)
- Support for various deployment scenarios and use cases
- Zero-touch when possible, low-touch when needed

Plug and Play Server

The Cisco Plug and Play (PnP) server is a central server that encodes the logic of managing and distributing deployment information (images, configurations, files, and licenses) for the devices being deployed. The server communicates with the agent that is installed on the device that supports the simplified deployment process, using a specific deployment protocol.

Figure 1: Simplified Deployment Server



The PnP server also communicates with proxy servers like deployment applications on smart phones and PCs, or other PnP agents acting as Neighbor Assisted Provisioning Protocol (NAPP) servers, and other types of proxy deployment servers like VPN gateways.

Cisco PnP supports redirection. For example, a PnP server can redirect a device to communicate with it directly after sending the bootstrap configuration through a NAPP server. If the PnP server is hosted by an enterprise leveraging a cloud-based deployment service provided by Cisco, the device discovers and communicates with the Cisco cloud-based deployment service for initial deployment. After that, the PnP server redirects the device to the customer's deployment server.

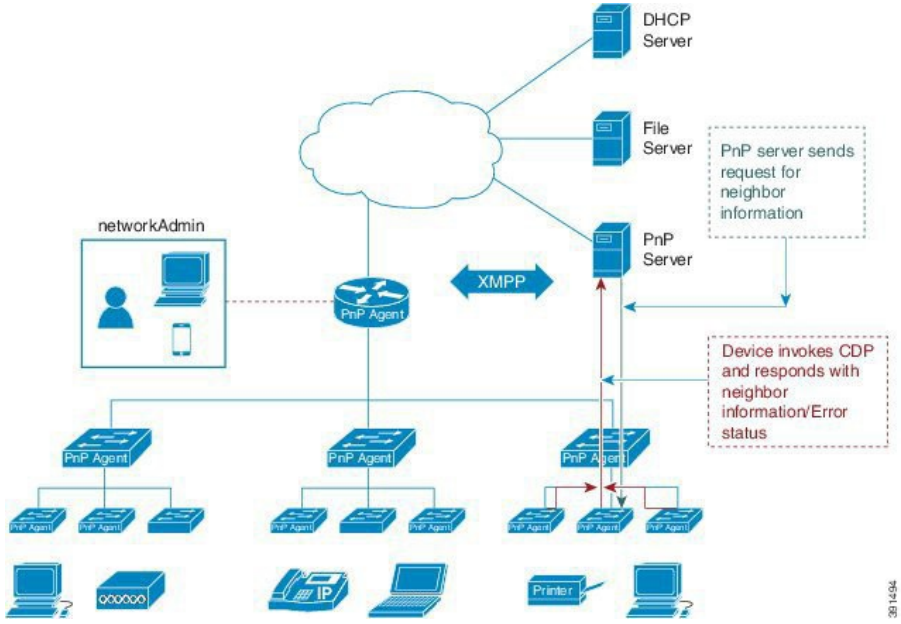
In addition to communicating with the devices, the server interfaces with a variety of external systems like Authentication, Authorizing, and Accounting (AAA) systems, provisioning systems, and other management applications.

How Cisco Plug and Play Works on Cisco Devices

The following steps detail the Cisco Plug-n-Play (PnP) deployment on Cisco devices:

1. The Cisco device that has the PnP agent installed on it, contacts the PnP server requesting an action by sending its unique device identifier (UDI) along with a request for work.
2. The PnP server, if it has an actionable step for the device, sends back a work request indicating the kind of action that the PnP agent must perform. For example, image installation or configuration upgrade.
3. When the PnP agent receives the work request, it executes the step and sends back a reply to the PnP server indicating the status of the step.

Figure 2: Network Topology of Cisco Plug-n-Play Deployment



Plug and Play Agent Initialization Scenarios

The Cisco Plug and Play (PnP) agent is enabled by default on your device. The PnP agent can be initiated on a device in the following ways:

Automatically Triggering PnP on Devices with No Startup Configuration

New Cisco devices are shipped to customers with no startup configuration file in the NVRAM of the devices. When a new device is connected to a network and powered on, the absence of a startup configuration file on the device automatically triggers the Cisco Plug and Play (PnP) agent to discover the PnP server IP address.

Figure 3: Workflow for PnP Trigger with no Startup Configuration

Initializing PnP Agent using the CLI

Network administrators can use the command line interface (CLI) to initiate the Plug-n-Play (PnP) agent process at any time. By configuring a PnP profile through the CLI, a network administrator can start and stop the PnP agent on a device. When the PnP profile is configured using the CLI, the device starts the PnP agent process, which in turn, initiates a connection with the PnP server using the IP address in the PnP profile.

Figure 4: Workflow for PnP Trigger with CLI Configured PnP Profile

Prerequisites for Cisco Plug and Play

- Deploy the discovery mechanism, either a DHCP server discovery process or a Domain Name Server (DNS) discovery process, before you launch the PnP agent.
- Configure the DHCP server or the DNS server before deploying the PnP agent.
- Ensure that the PnP agent is able to reach the PnP server.

- The PnP agent enforces the PnP server to send user credentials for every request. Cisco recommends the usage of HTTP secure (HTTPS) protocol.

Limitations and Guidelines

- Cisco Plug-n-Play (PnP) agent facilitates HTTP, Extensible Messaging and Presence Protocol (XMPP) and HTTP secure (HTTPS) transport based communication with the PnP server.
- HTTPS cannot be used on platforms where crypto-enabled images are not supported. not use Secure Sockets Layer [SSL] or Transport Layer Security [TLS] protocols if crypto-enabled images are used).
- You cannot create VLANs using PnP configuration push with default VTP mode as server. Use EEM applet to push the configuration with **vtp mode transport** command as part of the configuration.
- Cisco Network Plug and Play supports devices using VLAN 1 by default. To use a VLAN other than VLAN 1, adjacent upstream devices must use supported releases and configure the global **pnp startup vlan x** command on the upstream device, to apply this configuration to the Plug and Play device:. When you execute this command on an adjacent upstream device, the VLAN membership does not change on that device. However, all the active interfaces on the upcoming Plug and Play device are changed to the specified VLAN. This guideline applies to both routers and switches.

Note

When you use the non-VLAN 1 feature, ensure that all the neighboring switch devices are running Cisco IOS XE Release 3.6.3 and not the 3.6.0, 3.6.1, or 3.6.2 releases. For more information about related caveat CSCut25533 that exists in these previous releases, see the Caveats section in the Release Notes for Cisco Network Plug and Play.

Cisco Plug and Play Deployment Scenarios

When the device boots, the absence of any startup configuration on the NVRAM triggers the PnP discovery agent to acquire the IP address of the PnP server. In order to acquire the IP address of the PnP server, the PnP agent goes through one of the following discovery mechanisms:

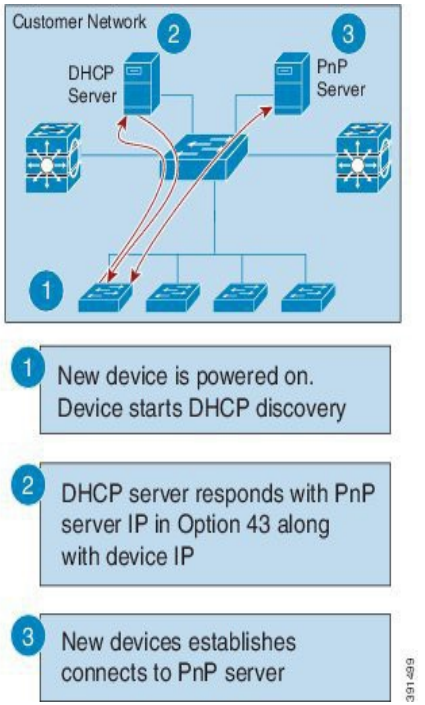
- PnP discovery through DHCP server
- PnP discovery through DHCP snooping
- PnP discovery through DNS lookup
- PnP proxy for layer 2 and layer 3 devices
- PnP deployment application

Plug and Play Discovery through DHCP Server

Device with no startup configuration in the NVRAM triggers the Cisco Plug and Play (PnP) agent to initiate a DHCP discovery process which requests an IP address from DHCP server required for the device. The DHCP server can be configured to insert additional information using the vendor-specific option 43 upon receiving option 60 from the device with the string 'cisco pnp'. This is to pass on the IP address or hostname of the PnP server to the requesting device.

When the DHCP response is received by the device, the PnP agent extracts the option 43 from the response to get the IP address or the hostname of the PnP server. PnP agent then uses this IP address or hostname to communicate with the PnP server.

Figure 5: DHCP Discovery Process for PnP s

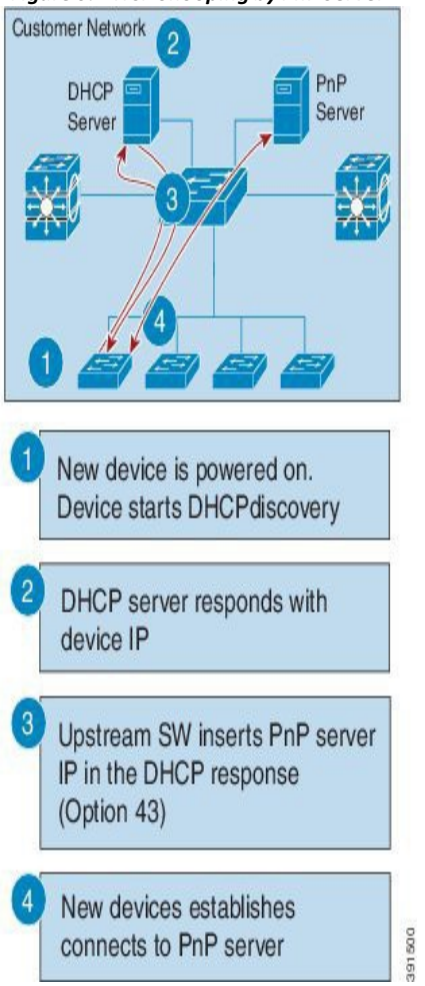


Plug and Play Discovery through DHCP Snooping

If a third party DHCP server cannot be configured to insert any vendor specific options, an existing Plug and Play (PnP) enabled device can be configured to snoop into the DHCP response and insert PnP specific option 43 with the PnP server IP address.

Before inserting DHCP option 43, the snooping agent verifies if the DHCP message is from a Cisco device in the network. The remaining DHCP discovery process is same as described in the previous section.

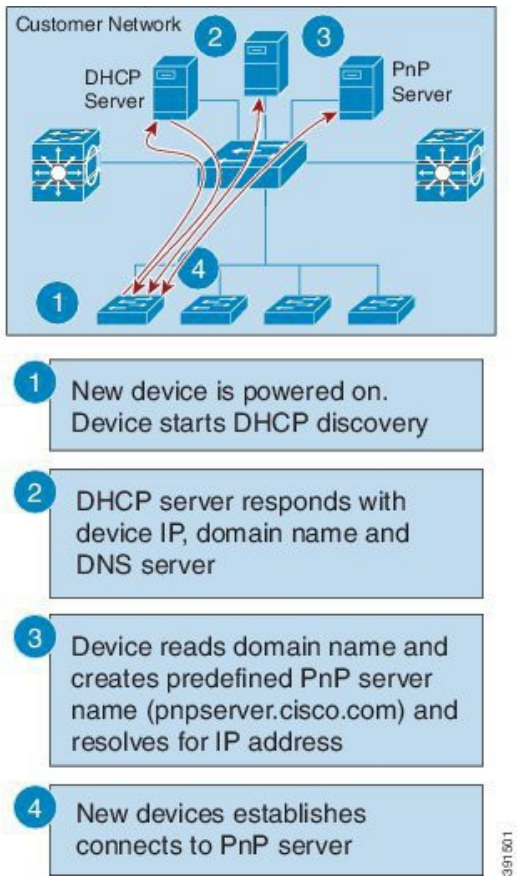
Figure 6: DHCP Snooping by PnP Server



Plug and Play Discovery through DNS Lookup

When the DHCP discovery fails to get the IP address of the Cisco Plug and Play (PnP) server, the agent falls back on Domain Name System (DNS) lookup method. The PnP agent then uses a preset deployment server name. The agent obtains the domain name of the customer network from the DHCP response and constructs the fully qualified domain name (FQDN). The following FQDN is constructed by the PnP agent using a preset deployment server name and the domain name information for the DHCP response, <*pnpservername*>.*cisco.com*. The agent then looks up the local name server and tries to resolve the IP address for the above FQDN.

Figure 7: DNS Lookup for deployment.customer.com



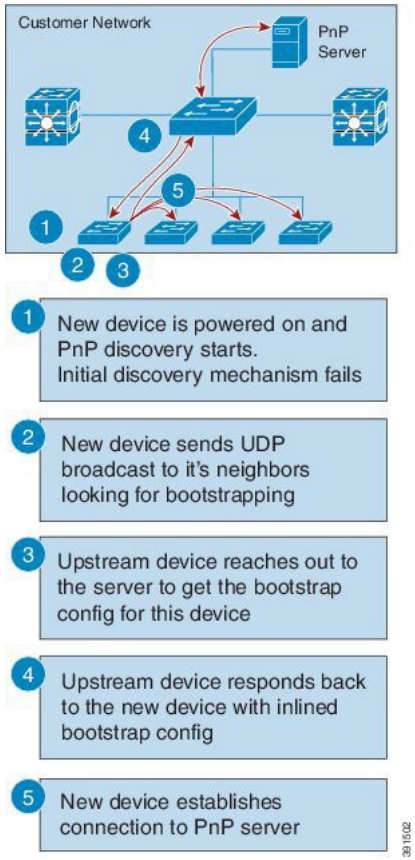
Plug and Play Proxy Server for Layer 3 and Layer 2 Devices

In the absence of DHCP or Domain Name System (DNS) servers, an existing up and running Cisco Plug and Play (PnP) enabled device in the neighborhood network can be configured to act as a PnP Neighbor Assisted Provisioning Protocol (NAPP) server.

The NAPP server is part of PnP discovery phase. This server is invoked when the PnP autonomic networking based discovery, DHCP, DNS, Cisco cloud service discovery mechanisms fail to connect to the PnP server.

This device listens to a specific port for any incoming PnP messages. The Cisco device which is trying to come up as a PnP device sends a UDP broadcast message to its network every 30 min for ten times. Hence, if the device does not receive a response, the broadcasts stop after 300 min.

Figure 8: DNS Looup for Layer 3 and Layer 2 Dev



When the device hosting the proxy server process receives the incoming broadcasts, it verifies the version field in the request and forwards the request to the PnP server if version validation is successful. The proxy server process also caches the unique device identifier (UDI) of the requesting client coming in via incoming datagram before forwarding the request to PnP server.

Upon receiving the configlet datagram from PnP server, the proxy server validates UDI in the incoming datagram with the entries in the UDI cache. If validation is successful, proxy server process broadcasts the datagram to a specific port number reserved for the proxy client processes to receive datagrams.

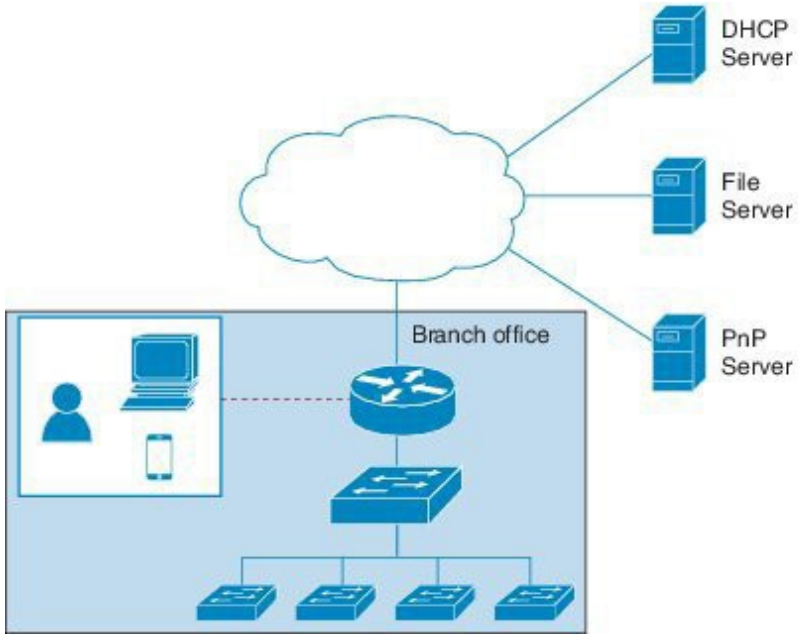
Upon receiving the datagrams, devices running proxy client processes, parse the incoming datagram for the target UDI. If the target UDI in the datagram matches the UDI of the device, proxy client process proceeds with framing, error control and configuring the configlet.

If the target UDI in the datagram fails to match UDI of the device, the packet is dropped.

Plug-n-Play Agent Deployment using a Deployment Application

Alternatively, your network administrator can manually configure your device using a deployment application running on their computer or on a smart phone. The computer or the smart phone can be connected to the device via USB or an Ethernet cable.

Figure 9: Manually Configured PnP Agent



Configuring Cisco Plug and Play

Perform the following steps to configure Cisco Plug-n-Play (PnP) on your device:

Configuring Cisco Plug and Play Agent Profiles

Perform the following steps to create a Cisco Plug and Play (PnP) agent profile:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.

Step 4	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Plug and Play Agent Devices

Perform the following steps to create a Cisco Plug and Play (PnP) agent device:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.

	Command or Action	Purpose
Step 4	**device** {**username** *username* } {**password** {**0** | **7**} *password*} **Example:** `Device(config-pnp-init)# device username sjohn password 0 Tan123`	Configures the PnP agent on the device. • Establishes a username and password based authentication system. • *username*—User ID • *password*—Password that a user enters • **0**—Specifies that an unencrypted password or secret (depending on the configuration) follows. • **7**—Specifies that an encrypted (hidden) password follows.
Step 5	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Plug and Play Reconnect Factors

Perform the following steps to configure the time to wait before attempting to reconnect a session in either fixed-interval-backoff, exponential-backoff, or random-exponential-backoff mode:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-`	Creates a PnP agent profile and enters the PnP profile initialization mode. String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**reconnect** [*pause-time* [*exponential-backoff-factor* [**random**]]] **Example:** `Device(config-pnp-init)# reconnect 100 2 random`	Specifies the time for the PnP agent initiator profile to wait before attempting to reconnect a session. • The pause-time value is the time to wait, in seconds, before attempting to reconnect after a connection is lost. The range is from 1 to 2000000. The default is 60. Exponential backoff factor value is the value that triggers the reconnect attempt exponentially. The range is from 2 to 9.

Step 5	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Cisco Plug and Play HTTP Transport Profiles

Perform the following steps to create a HTTP transport profile of the Plug-n-Play (PnP) agent manually on a device.

Both IPv4 and IPv6 addresses can be used for PnP server IP configuration. Alternately, a hostname can also be used in the configuration to connect to the PnP server. Every profile can have one primary server and a backup server configuration. The PnP agent attempts to initiate a connection with the primary server first and if it fails, it will try the backup server. If the backup server fails, the PnP agent will attempt to connect to the primary server again. This will continue until a connection is established with one of the servers.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**transport http host** *host-name* [**port** *port-number*] [**source** *interface-type*] **Example:**	Creates a HTTP transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed. • The value of the host specifies the host name, port, and source of the server.
Step 5	**transport http ipv4** *ipv4-address* [**port** *port-number*] [**source** *interface-type*] **Example:**	Creates a HTTP transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed.
Step 6	**transport http ipv6** *ipv6-address* [**port** *port-number*] [**source** *interface-type interface-number*] **Example:**	Creates a HTTP transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.
Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Cisco Plug and Play HTTPS Transport Profiles

Perform the following steps to create a HTTP Secure (HTTPS) transport profile of the Cisco Plug and Play (PnP) agent manually on a device.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**transport https host** *host-name* [**port** *port-number*][**source** *interface-type*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# transport https host example.com port 231 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates a HTTPS transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed. • The value of *localcert* specifies the trustpoint used for client-side authentication during the transport layer security (TLS) handshake. • The value of *remotecert* specifies the trustpoint used for server certificate validation. **Note** Configure the trustpoint-name using the **crypto pki trustpoint** command.

Step 5	**transport https ipv4** *ipv4-address* [**port** *port-number*][**source** *interface-type*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# transport https ipv4 10.0.1.0 port 221 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates a HTTPS transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed.
Step 6	**transport https ipv6** *ipv6-address* [**port** *port-number*][**source** *interface-type interface-number*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# transport https ipv6 2001:DB8:1::1 port 331 source gigabitEthernet 0/0/1 localcert abc remotecert xyz`	Creates a HTTPS transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.
Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Cisco Plug and Play XMPP Transport Profiles

Perform the following steps to create a Extensible Messaging and Presence Protocol (XMPP) transport profile of the Cisco Plug and Play (PnP) agent manually on a device.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.

Step 4	**transport xmpp socket** {**host** *host-name* | **ipv4** *ipv4-address*| **ipv6** *ipv6-address*} {**port** *port-number* } {**source** *interface-type interface-number* } {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# transport xmpp socket host example.com port 231 sasl plain server-jid cisco123`	Creates an XMPP transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed.
Step 5	**transport xmpp starttls** {**host** *host-name* | **ipv4** *ipv4-address*| **ipv6** *ipv6-address*} {**port** *port-number* } {**source** *interface-type interface-number* } {**localcert** *trustpoint-name* } {**remotecert** *trustpoint-name* } {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# transport xmpp starttls ipv4 10.0.1.0 port 221 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates an XMPP transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed. • The value of *localcert* specifies the trustpoint used for client-side authentication during the transport layer security (TLS) handshake. • The value of remotecert specifies the trustpoint used for server certificate validation.
Step 6	**transport xmpp tls** {**host** *host-name* | **ipv4** *ipv4-address*| **ipv6** *ipv6-address*} {**port** *port-number* } {**source** *interface-type interface-number*} {**localcert** *trustpoint-name*} {**remotecert** *trustpoint-name* } {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# transport xmpp tls ipv6 2001:DB8:1::1 port 221 source gigabitEthernet 0/0/0`	Creates an XMPP transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.

Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Backup Cisco Plug and Play Devices

Perform the following steps to create a backup profile and to enable or disable Cisco Plug and Play agent manually on a device:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.

Step 4	**backup device** {**username** *username* } {**password {0 | 7}** *password*} **Example:** `Device(config-pnp-init)# backup device username sjohn password 0 Tan123`	Configures the PnP agent backup profile on the device. • Establishes a username and password based authentication system. • *username*-User ID • *password*-Password that a user enters • 0—Specifies that an unencrypted password or secret (depending on the configuration) follows. • 7—Specifies that a hidden password follows.
Step 5	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Backup Cisco Plug and Play Reconnect Factors

Perform the following steps to configure backup reconnection of the Cisco Plug and Play (PnP) agent to the server in either fixed-interval-backoff, exponential-backoff, or random-exponential-backoff manner :

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**backup reconnect** [*pause-time* [*exponential-backoff-factor* [**random**]]] **Example:** `Device(config-pnp-init)# backup reconnect 100 2 random`	Specifies the time for the PnP agent initiator profile to wait before attempting to reconnect a session. • The pause-time value is the time to wait, in seconds, before attempting to reconnect after a connection is lost. The range is from 1 to 2000000. The default is 60. • Exponential backoff factor value is the value that triggers the reconnect attempt exponentially. The range is from 2 to 9.
Step 5	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Backup Cisco Plug and Play HTTP Transport Profile

Perform the following steps to create a backup HTTP transport profile of the Cisco Plug and Play (PnP) agent manually on a device.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**backup transport http host** *host-name* [**port** *port-number*] [**source** *interface-type*] **Example:** `Device(config-pnp-init)# backup transport http host hostname-1 port 1 source gigabitEthernet 0/0/0`	Creates a backup HTTP transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed. • The value of the host specifies the host name, port, and source of the server. • The value of the port-number specifies the port that is used. • The value of the interface-type specifies the interface on which the agent is connected to the server.

Step 5	**backup transport http ipv4** *ipv4-address* [**port** *port-number*] [**source** *interface-type*] **Example:** `Device(config-pnp-init)# backup transport http ipv4 10.0.1.0 port 221 source gigabitEthernet 0/0/0`	Creates a backup HTTP transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed.
Step 6	**backup transport http ipv6** *ipv6-address* [**port** *port-number*] [**source** *interface-type interface-number*] **Example:** `Device(config-pnp-init)# backup transport http ipv6 2001:DB8:1::1 port 331 source gigabitEthernet 0/0/1`	Creates a backup HTTP transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.
Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Backup Cisco Plug and Play HTTPS Transport Profile

Perform the following steps to create a backup HTTPS transport profile of the Cisco Plug and Play (PnP) agent manually on a device.

	Command or Action	Purpose

Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.
Step 4	**backup transport https host** *host-name* [**port** *port-number*][**source** *interface-type*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# backup transport https host example.com port 231 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates a HTTPS backup transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed. • The value of *localcert* specifies the trustpoint used for client-side authentication during the transport layer security (TLS) handshake. • The value of *remotecert* specifies the trustpoint used for server certificate validation.

Step 5	**backup transport https ipv4** *ipv4-address* [**port** *port-number*][**source** *interface-type*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# backup transport https ipv4 10.0.1.0 port 221 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates a HTTPS backup transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed.
Step 6	**backup transport https ipv6** *ipv6-address* [**port** *port-number*][**source** *interface-type interface-number*][**localcert** *trustpoint-name*][**remotecert** *trustpoint-name*] **Example:** `Device(config-pnp-init)# backup transport https ipv6 2001:DB8:1::1 port 331 source gigabitEthernet 0/0/1 localcert abc remotecert xyz`	Creates a HTTPS backup transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.
Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Backup Cisco Plug and Play XMPP Transport Profile

Perform the following steps to create a backup Extensible Messaging and Presence Protocol (XMPP) transport profile of the Cisco Plug and Play (PnP) agent manually on a device.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp profile** *profile-name* **Example:** `Device(config)# pnp profile test-profile-1`	Creates a PnP agent profile and enters the PnP profile initialization mode. • String of alphanumeric characters that specify a name for the PnP agent profile. Profile names cannot be duplicated.

Step 4	**backup transport xmpp socket** {**host** *host-name* | **ipv4** *ipv4-address*|**ipv6** *ipv6-address*} {**port** *port-number*} {**source** *interface-type interface-number*} {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# backup transport xmpp socket host example.com port 231 sasl plain server-jid cisco123`	Creates an XMPP transport configuration for the PnP agent profile based on the hostname of the server on which the PnP agent is deployed.
Step 5	**backup transport xmpp starttls** {**host** *host-name* | **ipv4** *ipv4-address*|**ipv6** *ipv6-address*} {**port** *port-number*} {**source** *interface-type interface-number*} {**localcert** *trustpoint-name*} {**remotecert** *trustpoint-name*} {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# backup transport xmpp starttls ipv4 10.0.1.0 port 221 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates an XMPP transport configuration for the PnP agent profile based on the IPv4 address of the server on which the PnP agent is deployed. • The value of *localcert* specifies the trustpoint used for client-side authentication during the transport layer security (TLS) handshake. • The value of *remotecert* specifies the trustpoint used for server certificate validation.
Step 6	**backup transport xmpp tls** {**host** *host-name* | **ipv4** *ipv4-address*|**ipv6** *ipv6-address*} {**port** *port-number*} {**source** *interface-type interface-number*} {**localcert** *trustpoint-name*} {**remotecert** *trustpoint-name*} {**sasl plain server-jid** *xmpp-jabber-id*} **Example:** `Device(config-pnp-init)# backup transport xmpp tls ipv6 2001:DB8:1::1 port 221 source gigabitEthernet 0/0/0 localcert abc remotecert xyz`	Creates an XMPP transport configuration for the PnP agent profile based on the IPv6 address of the server on which the PnP agent is deployed.

Step 7	**end** **Example:** `Device(config-pnp-init)# end`	Exits the PnP profile initialization mode and returns to privileged EXEC mode.

Configuring Cisco Plug and Play Agent Tag

Perform the following step to create Cisco Plug and Play (PnP) agent tag information:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**pnp tag** *tag-name* **Example:** `Device(config)# pnp tag xyz`	Use the **pnp tag** command to configure the tag for the device. The neighboring Cisco devices will receive this tag information through Cisco Discovery Protocol (CDP). **Note** If there is an existing tag for the device, the tag name can be only changed when the xml schema is sent by the PnP server to change the tag name. The tag name cannot be overwritten. • String of alphanumeric characters that specify a name for the PnP agent tag.

Step 4	**end** **Example:** `Device(config)# end`	Exits the global configuration mode and returns to privileged EXEC mode.

Troubleshooting

The **show pnp tech-support** command can be used to verify the existing configuration. The sample output is as given below:

```
------------------ show pnp summary ------------------

PnP Schema Version: 1.0, Baseline Tracking: rel14.1.150612
Device UDI: PID:WS-C3650-48PD,VID:V01,SN:FDO1732Q00R
UDI Checking: Yes
Security Enforced: Yes, PostReloadPriv'd Profile: N/A
SUDI Certificate: N/A
Device SUDI: N/A

------------------ show pnp udi tracking ---------

Best UDI:[PID:WS-C3650-48PD,VID:V01,SN:FDO1732Q00R]
Good UDI:[PID:WS-C3650-48PD,VID:A0,SN:FDO1732Q00R]
Incomplete UDI:[-]
UDI by Master Registry:[PID:WS-C3650-48PD,VID:A0,SN:FDO1732Q00R]
UDI by Entity MIBS:[PID:WS-C3650-48PD,VID:V01,SN:FDO1732Q00R]
UDI by Platform Registry:[PID:WS-C3650-48PD,VID:,SN:FDO1732Q00R]

-------------- show pnp config tracking ---------------

Config Monitor: Off, Switched: 2
Config Control Level:[All-Check], Last-ConfControl:[All-Check]
Config Retry: 300, Interval: 1000 ms
Config Reserved By:[-], Last-ConfReserve:[-]
Startup Config: Found, Write Started: 0, Done: 0, PID: 0, Last-PID: 0
Running Config: Not Locked, Safe Now: -, CLI Changed: 0, Bulk Count: 0, Last Delta: 0, PID: 0, Last-PID: 0
HA Present: Yes, Registry: Yes, Config Sync: -
Standby Notify Hot: 0, Cold: 0
```

In the above output, **show pnp config tracking** can be used to verify if any non pnp feature is changing the configuration in the background.

Viewing Debug information

To run the debugging on the Cisco Plug and Play (PnP) server, start the server, configure the PnP profile and PnP transport. That is, start the service interaction between PnP agent and PnP server. Capture the debugs by executing the **debug pnp service** command.

Resources and Support Information

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service.

TOMORROW
starts here.