Deploy Router Using Bootz

With the Bootz process, you can securely and seamlessly provision network devices accurately within minutes and without any manual intervention.

Table 1: Feature History Table

Feature	Release Information	Feature Description
Provisioning Using Bootz Process	Release 7.11.1	This feature allows devices in the network to establish a secure connection with the remote Bootz server and authenticate information using a three-step validation process. This process involves validating the network device, the Bootz server, and the onboarding information thereby mitigating security risks and preventing malicious actions during remote provisioning.

Unlike the Secure ZTP process, which relies on vendor-specific definitions for bootstrapping a device, the Bootz process offers a specification that outlines data elements in a vendor-agnostic manner. It also details the necessary operations at turn-up time, integrating them into the boot process.

Also, the bootstrap request in the Bootz process includes the unique identifier or serial number for each node as opposed to the Secure ZTP process where the bootstrap request does not include serial numbers. The Bootz server returns the signed onboarding information with ownership voucher and owner certificate for the requested serial number of the device.

Supported Bootz Versions

This table provides the Bootz versions supported in each release. The Bootz Bootstrap server must be compatible with the respective Bootz version.

Table 2: Bootz Versions

Release	Version with File Path
Release 24.3.1	`openconfig/bootz v0.3.1`
Release 24.2.1	`openconfig/bootz v0.1.1-0.20231106050618-8d6e2559f803`
Release 24.1.1	`openconfig/bootz v0.0.0-20230809153947-e6bc0be82dd8`
Release 7.11.1	`openconfig/bootz v0.1.0`

Components used in the Bootz Process

These components are part of the Bootz process.

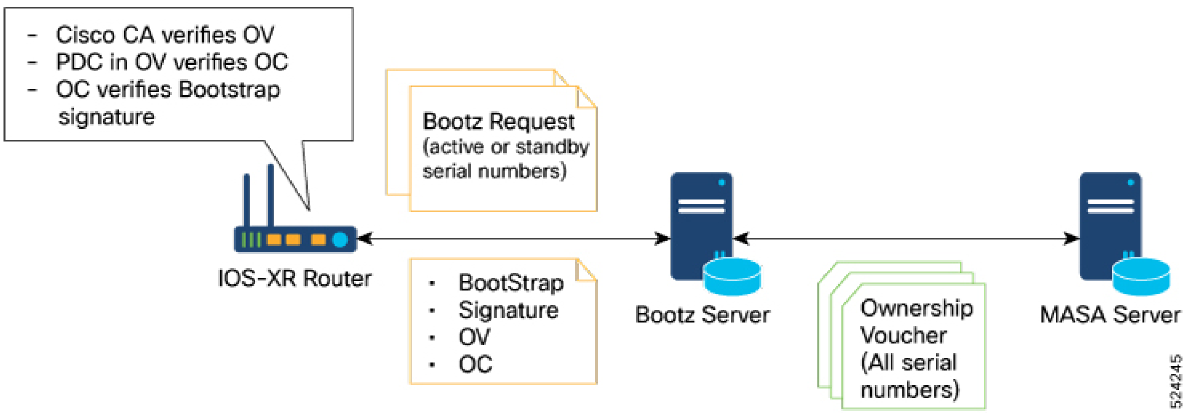
- **Onboarding Device (Router):** A router is a Cisco device that you want to provision and connect to your network. Bootz is supported only on platforms that have *Hardware TAM*[1] support.
- **DHCP Server:** The DHCP server provides the URL where the Bootz process can access the bootstrapping information.
- **MASA Server:** You can generate and store the ownership voucher in the MASA server. The MASA server sends the ownership voucher to the Bootz server so that the Bootz process validates the device and establishes device ownership.
- **Bootz Bootstrap Server:** A Bootz Bootstrap server is any gRPC server used as a Bootz bootstrapping data source. For example, Google Proto. The Bootz Bootstrap server is complaint with Openconfig Bootz standards.

Note Bootz only supports a single name-server. As a result, when the DHCP server has more than one server address configured, Bootz fails to apply the server configuration.

The Bootz server contains these artifacts:

- **Cisco IOS XR software images:** You can download Cisco images, SMU, and patches from the Cisco Support & Downloads page.
- **Bootstrapping Data:** It is a collection of data that you have created and uploaded to the Bootz server. The router obtains this data from the Bootz server during the provisioning process.

1 A secure storage device that stores the customer certificates and Cisco's internal secure data like trust anchors, SUDI certificates, secure flags, and other security information.



Onboard Devices Using Bootz Workflow

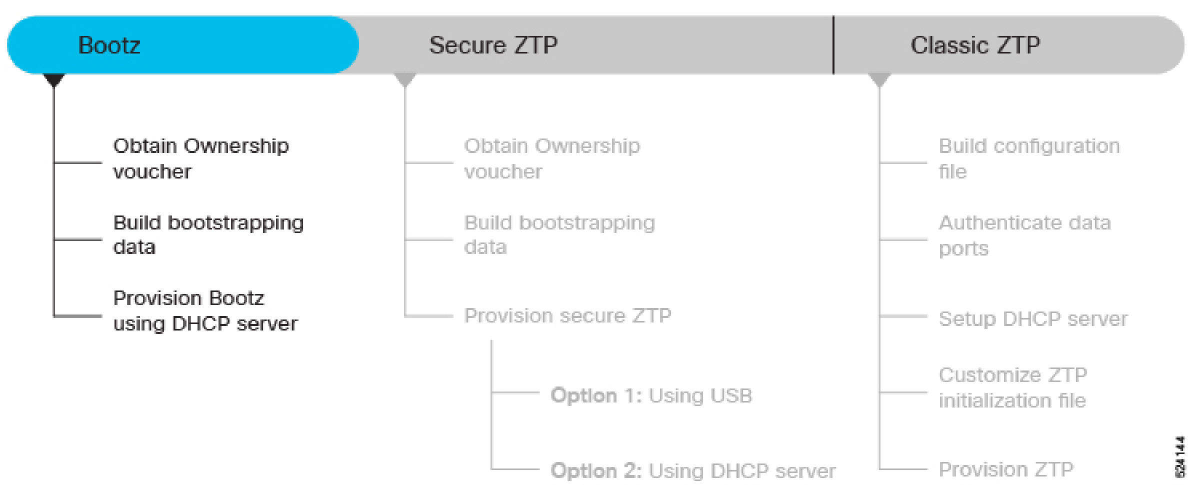
The Cisco IOS XR software supports Bootz provisioning capabilities. The Bootz process uses the Google Remote Procedure Call (gRPC) protocol for fetching information from a remote server.

The Bootz workflow performs these validations to onboard the remote devices securely.

1. **Router Validation**: The Bootz server authenticates the router before providing the bootstrapping data.
2. **Server Validation**: The router in turn validates the Bootz server and ensures that the onboarding is performed for the correct network. Once it is validated, the Bootz server sends the bootstrapping data (for example, a YANG data model) or artifact to the router.
3. **Artifact Validation**: The router validates the bootstrapping data or artifacts received from the Bootz server.

This figure provides the Bootz workflow and the processes involved in the workflow. The sections that follow describe these processes in detail.

Figure 1: Bootz Workflow



Obtain Ownership Voucher

The ownership voucher is used to identify the owner of the device by verifying the owner certificate stored in the device.

How to obtain Ownership Voucher

These steps help you obtain the ownership voucher from Cisco:

1. Contact Cisco Support.

2. Provide these information in your request to Cisco.

    - **Pinned Domain certificate (PDC):** PDC is an X.509 v3 certificate structure that uses Distinguished Encoding Rules (DER). The router uses this certificate to trust a public key infrastructure for verifying a domain certificate supplied to the router separately in the bootstrapping data. This certificate could be an end-entity certificate, including a self signed entity.

    - Purchase order details with the serial numbers of the routers.

    Sample Request:

    ```
    {
       "expires-on": "2016-10-21T19:31:42Z",
       "assertion": "verified",
       "serial-number": "JADA123456789",
       "idevid-issuer": "base64encodedvalue==",
       "pinned-domain-cert": "base64endvalue==",
       "last-renewal-date": "2017-10-07T19:31:42Z"
    }
    ```

3. Cisco generates the ownership voucher in .vcj format (Example: DCA213140YX.vcj) and sends the voucher in response to your request.

Build Bootstrapping Data

Steps to build the bootstrapping data:

1. Create and upload the bootstrapping data to the gRPC server or Bootz bootstrap server.

2. The router sends a bootstrap request with these artifacts to the Bootz server.

    - Serial number of the control card or line card

    - Software image to download and install

    - Bootloader Password for the device

    - Certificate used to validate the bootstrap server

    - Bootstrap server configuration information such as server credentials, path information, authentication information, and certificates

    For the request message format, see the Bootstrap Request Message.

3. The Bootz server returns the listed bootstrapping data in its response to the router. The router receives these data during the provisioning process.

    - **Signed Bootstrap Response:** Each bootstrap response contains the onboarding information for:

      For the response message format, see the Bootstrap Response Message for a single card.

    - **Owner Certificate:** The owner certificate is installed on the router with your organization's public key. The router uses this public key in the owner certificate to verify the signature in the signed bootstrap response artifact.

    - **Ownership Voucher:** The ownership voucher is used to identify the device owner by verifying the owner certificate stored in the device. Cisco generates and supplies the ownership voucher in response to your request containing the PDC and device serial numbers. For more information, see Obtain Ownership Voucher.

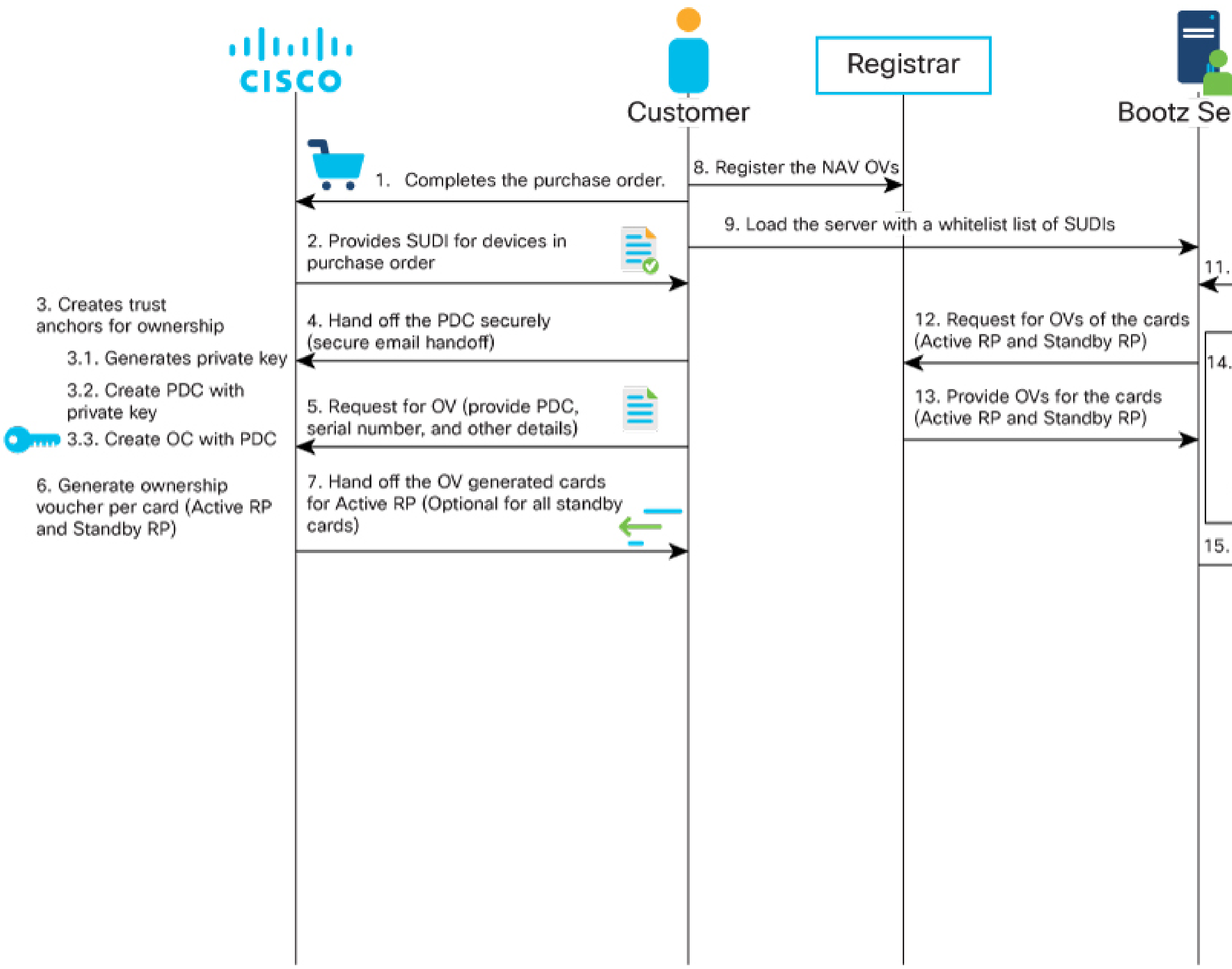
4. When the router obtains the onboarding information from the Bootz server, the router reports the bootstrapping progress to the Bootz server using the API calls.

Provision Bootz Using DHCP Server

When you boot the device, the Bootz process initiates automatically on a device without prior configuration. During the process, the router receives the details of the configuration file from the DHCP server.

This figure illustrates the end-to-end sequence of the Bootz process:

Figure 2: End-to-end sequence of the Bootz process



Before you begin

As part of the initial setup for secure ZTP, the network administrator:

- Ensures to enable secure ZTP on the router using the **ztp secure-mode enable** command and reload the router.
- Contacts Cisco Support and follows the steps in Obtain Ownership Voucher to obtain a voucher from Cisco.

Procedure

Step 1 Upload the listed bootstrapping data to the Bootz server. Refer to your vendor documentation as the upload procedure may vary from server to server.

- Cisco IOS XR software images

Note Download Cisco images, SMU, and patches from the Cisco Support & Downloads page.

- Serial numbers of the routers to be onboarded
- Owner certificates
- Pinned Domain Certificate (PDC)
- Ownership vouchers

Step 2 Set up the DHCP server to provide the redirect URL to the router:

Before triggering the secure ZTP process, configure the DHCP server so that it provides the location of the IOS-XR image to the router. For information about how to configure the DHCP server, see your DHCP server documentation.

Configure these parameters in the DHCP server:

- `option-code`: Use one of these DHCP SZTP redirect option parameters in the `option-code` setting.
  - OPTION_V4_SZTP_REDIRECT (143): DHCP v4 code for IPv4.
  - OPTION_V6_SZTP_REDIRECT (136): DHCP v6 code for IPv6.
- `option-length`: Provide the option length in octets.
- `bootstrap-servers`: A list of servers. The onboarding device contact these servers for the bootstrapping data.

```
"bootz://<ip-address-or-hostname>[:<port>]<endpoint>"
```

Example: `option dhcp6.bootstrap-servers code 136 = text;`

Step 3 Power on the router.

This procedure provides the high-level workflow of the Bootz process:

a. When you boot the device with an IOS-XR image, the secure ZTP process verifies if the secure ZTP mode (`secure-ztp mode`) is enabled. If not enabled, the device boots normally.

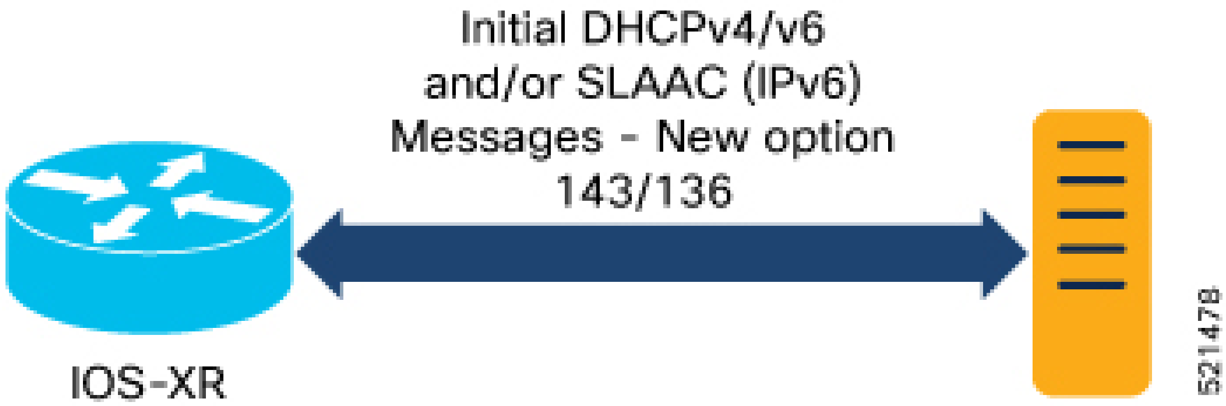
Note When `secure-ztp mode` is enabled, the ZTP process accepts only the `secure-redirect-URL` and ignores the presence of the boot file name option from the DHCP response.

b. **DHCP discovery**:

1. The router initiates a DHCP request to the DHCP server.
2. The DHCP server responds with a DHCPv4 143 address option (for IPv4 addressing) or a DHCPv6 136 option (for IPv6 addressing).

Note URLs to access bootstrap servers for further configuration are listed in options 136 and 143.

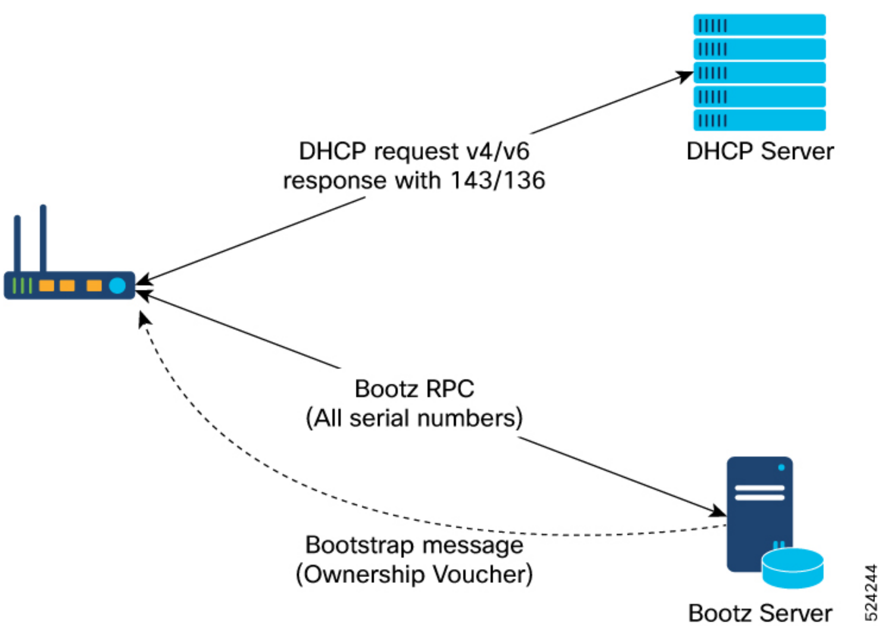
Figure 3: DHCP discovery



c. **Router and Bootz server validation**:

1. After receiving the URL from the DHCP server, the router initiates a gRPC connection to the Bootz server. The Bootz server IP address is obtained from the DHCP response.
2. The Bootz server authenticates the router before it provides the bootstrapping data.
3. After the Bootz server authenticates the router or the onboarding device, the router validates the Bootz server to ensure that the onboarding is performed for the correct network.

   After validating the Bootz server, the router sends the serial number for each control card or line card and other artifacts in its bootstrap request.
4. After its validation, the Bootz server sends the required artifacts along with the bootstrap response data to the router or the onboarding device.



d. **Ownership Voucher verification**:

The router receives the bootstrap response data that contains owner certificate, ownership voucher for each serial number, and the details of the image upgrade, if any.

Bootstrap response data includes the following:

- Image path
- Image version
- Trust anchor
- Boot configuration
- GNSI artifacts

These artifacts come from the Bootz server as a bootstrap response gRPC message. The router verifies the ownership voucher by validating its signature to one of its preconfigured trust anchors and downloads the image. When the router obtains the onboarding information, it reports the bootstrapping progress to the Bootz server.

e. **Artifact Validation**:

The router validates the artifacts received from the Bootz server as follows:

1. The device extracts the `pinned-domain-cert` node, an X.509 certificate from the ownership voucher to verify the owner certificate.
2. The device authenticates the owner certificate by performing the X.509 certificate path verification process on the trusted certificate.
3. Finally, the device verifies whether the artifact is signed by the validated owner certificate.

f. **Provision the device**:

1. The device first processes the boot image information.
2. Executes the script and then onboards the artifacts received from the Bootz server.

g. After the onboarding process is completed, the network device is operational.

Bootz Workflow for Standby RP

Table 3: Feature History Table

Feature	Release Information	Feature Description
Bootz Workflow for Standby RP	Release 24.3.1	*Introduced in this release on: Modular Systems (8800 [LC ASIC: Q100, Q200, P100])* This feature enables the Bootz workflow to achieve full-system onboarding for devices with both active and standby Route Processors (RPs). In earlier releases, the Bootz workflow only supported onboarding for devices with an active RP. With this enhancement, the Bootz workflow can now detect faulty or tampered standby cards that are inserted dynamically during or after the active RP Bootz process. It does this by verifying the ownership voucher (OV) of the other cards during the initial Bootz process for the active RP. Faulty cards can be shut down to prevent security threats during remote provisioning, ensuring smooth network operation. This feature allows Bootz workflow to validate the standby RP as part of the active RP Bootz process. This feature introduces the **ztp bootz-server** command. This feature modifies the `Cisco-IOS-XR-ztp-cfg.yang`. (see GitHub, YANG Data Models Navigator)

Overview

The Bootz workflow now processes the ownership voucher (OV) for multiple control cards or line cards detected on the standby RP, either before the Bootz process for an active RP starts, during its progress, or after its completion.

With this new feature, the Bootz workflow:

- Allows dynamic insertion or replacement of standby control cards and line cards.

- Processes the ownership voucher (OV) bundle that the Bootz server sends to the router for the standby control cards or line cards.
- Performs the ownership verification of all standby control cards and line cards.

This feature allows you to configure the ZTP Bootz server with the **ztp bootz-server** command to store the server and vendor information received during the initial Bootz process for the active RP.

The router uses this configuration to communicate with the Bootz server and obtain the OV bundle (.tar file) for dynamically inserted or replaced standby control cards or line cards.

Prerequisites

- Configure the Bootz server to return the bootstrap data response message for all the serial numbers of the cards on the device with either the OV bundle or individual ownership voucher for each card.
- Include the **ztp bootz-server** configuration in the server's onboarding information or the vendor configuration information. This configuration is received from the Bootz server during the initial GetBootstrapDataRequest exchange for the active RP.
- Ensure that the routers running the Bootz client can process the OV bundle.

Restrictions

The dynamic Bootz workflow for the standby RP is triggered only if these conditions are met:

- Secure ZTP is enabled on the device.
- The Bootz process for the active RP is completed or not in progress.
- The Bootz server configuration from the initial Bootz process for active RP is available for dynamically inserted standby cards.

Use Cases

These use cases describe different scenarios where the standby RP cards are detected and the OV information is processed accordingly.

Use Case 1 - Standby Card Detected Before Bootz Process for Active RP Starts

If both the active RP and standby RP are detected during the initial boot process before the router communicates with the Bootz server:

- The router sends a bootstrap request to the Bootz server, including the serial numbers for both the active RP and standby RP.
- The Bootz server responds with the OV information for both the active RP and standby RP in its bootstrap response.
- If the response message for the initial bootstrap data request (for active RP) does not include an OV bundle for the standby RP, an additional bootstrap data request is triggered to fetch the OV information for the standby RP.

If there is no OV bundle in the bootstrap response for the initial bootstrap data request (for active RP),

Use Case 2 - Standby Card Detected During Bootz Process for Active RP

If the standby card is detected while the Bootz process for the active RP is in progress, the Bootz process for the standby RP is automatically triggered after the active RP's Bootz process completes.

In this scenario, the router uses the server information received during the initial Bootz process for the active RP, which you have configured using the **ztp bootz-server** command, to:

- Communicate with the Bootz server.
- Send the serial numbers for the dynamically inserted cards in its bootstrap request for standby RP to the Bootz server.
- Obtain the OV bundle (.tar file) from the Bootz server and process the OV for each card with a matching serial number. For more information about the OV bundle, see How the Router Obtains and Processes the OV Information.

The Bootz server, in turn, sends the OV information for the standby RP in its bootstrap response.

Use Case 3 - Standby Card Detected After Bootz Process for Active RP Completes

If the standby RP is detected after the initial Bootz process for active RP is completed, the Bootz process is triggered again automatically. The Bootz process is re-triggered if one of the following events occur:

- When a new card is inserted.
- When an existing card is replaced with another card.

Store the server information obtained during the initial Bootz process for active RP. As the secure ZTP workflow is not re-triggered for dynamically inserted standby control cards or line cards, the router uses this stored server information to communicate with the Bootz server and obtain the ownership vouchers for the newly inserted standby control cards or line cards.

Configuration to store server information obtained from the active RP Bootz process:

```
RP/0/RP0/CPU0:ios# config
RP/0/RP0/CPU0:ios(config)#ztp bootz-server ip 1.1.1.1 port 5000 trust-anchor
/misc/disk1/ta.cert
RP/0/RP0/CPU0:ios(config)# commit
```

Once the dynamic Bootz workflow for standby RP is triggered, the router communicates with the Bootz server using the server and vendor configuration information specified in the **ztp bootz-server** command. The Bootz server then sends the OV information for the dynamically inserted standby cards in its bootstrap response.

How the Router Obtains and Processes the OV Information

The router uses one of these methods to process the OV information that it has obtained from the Bootz server.

- **If the Bootz server is configured to send an OV bundle:**
  1. The Bootz server sends the OV bundle as a single tar file in the bootstrap response to the router for the RPs.
  2. The router running the Bootz client processes the tar file to verify individual ownership voucher for each serial number.
  3. The router loads the owner certificate and the ownership voucher on each card with a serial number matching the serial number included in the ownership voucher.

- **If the Bootz server is configured to send individual ownership vouchers rather than an OV bundle, the router:**
  1. Communicates with the Bootz server using the server information from the bootstrap response for active RP.
  2. Sends a new bootstrap request for the standby RP
  3. Obtains the ownership voucher information for the standby RP.