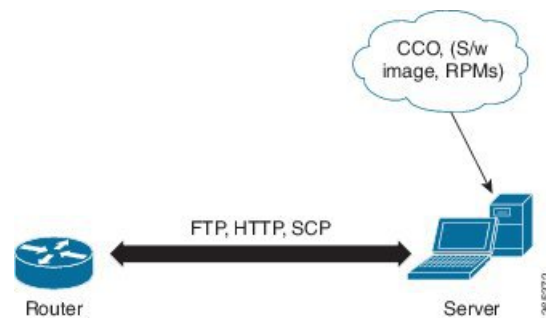# Manage Automatic Dependency

Flexible packaging supports automatic dependency management. While you update an RPM, the system automatically identifies all relevant dependent packages and updates them.

Figure 1: Flow for Installation (base software, RPMs and SMUs)

Until this release, you downloaded the software image and required RPMs from CCO on a network server (the repository), and used the **install add** and the **install activate** commands to add and activate the downloaded files on the . Then, you manually identify relevant dependent RPMs, to add and activate them.

With automatic dependency management, you need not identify dependent RPMs to individually add and activate them. You can execute new install command to identify and install dependent RPMs automatically.

The new commands are **install update** and **install upgrade**. The **install update** command identifies and updates dependent packages, but does not update the base package. The **install upgrade** command upgrades the base package.

> **Note**
> 1.
> 2.

The rest of this chapter contains these sections:

Update RPMs and SMUs

An RPM may contain a fix for a specific defect, and you may need to update the system with that fix. To update RPMs and SMUs to a newer version, use the **install add** command. When this command is issued for a particular RPM, the router communicates with the repository, and downloads and activates that RPM. If the repository contains a dependent RPM, the router identifies that dependent RPM and installs that too.

The syntax of the **install add** command is:

install add source *repository* **[rpm]**

Four scenarios in which you can use the **install add** command are:

- **When a package name is not specified**

  When no package is specified, the command updates the latest SMUs of all installed packages.

  install add source `[repository]`

- **When a package name is specified**

  If the package name is specified, the command installs that package, updates the latest SMUs of that package, along with its dependencies. If the package is already installed, only the SMUs of that package are installed. (SMUs that are already installed are skipped.)

- **When a package name and version number are specified**

  If a particular version of package needs to be installed, the complete package name must be specified; that package is installed along with the latest SMUs of that package present in the repository.

- **When an SMU is specified**

  If an SMU is specified, that SMU is downloaded and installed, along with its dependent SMUs.

Upgrade Base Software Version

You can upgrade to a newer version of the base software when it becomes available. To upgrade to the latest base software version, use the **install upgrade** command. With the upgrade of the base version, RPMs that are currently available on the router are also upgraded.

> **Note** SMUs are not upgraded as part of this process.

The syntax of the **install upgrade** command is:

install upgrade sourceinstall source *repository*

> **Note** VRF and TPA on dataport is not supported. If the server is reachable only through non-default VRF interface, the file must already be retrieved using ftp, sfp, scp, http or https protocols.

> **Note** Default routes (0.0.0.0/0) cannot be copied onto Linux due to TPA implementation.

You can use the **install upgrade** command when:

- **The version number is specified**

  The base software (.mini) is upgraded to the specified version; all installed RPMs are upgraded to the same release version.

  install upgrade source install source *[repository]* **version** *<version>* **asr9k-mini-x64-<version>.iso**

  For example,

  install source *repository* **version 7.0.1 asr9k-mini-x64-7.0.1.iso**

  You can also automatically fetch the .mini file and RPMs of the required release and proceed with the upgrade.

  install source *repository* **asr9k-mini-x64-7.0.1.iso**

Downgrade an RPM

After an RPM is activated, you may need to downgrade it by activating an RPM of a lower version. Use the **force** option with the **install activate** command to activate an RPM of a lower version.

The syntax of the command is: **install activate[rpm]force**

For example, to add and activate an RPM of a lower version, use the following steps.

Configuration

1. Download the lower version RPM to the .

   RPM currently active: mpls-2.0.0.0-r60011I

   RPM to be activated: mpls-2.0.0.0-r6006I

   install add source[repository] mpls-2.0.0.0-r6006I.rpm

2. Activate the downloaded RPM.

   install activatempls-2.0.0.0-r6006I.rpm force

   On activation, **mpls-2.0.0.0-r60011I.rpm** is automatically rendered inactive.

You can use the **show install active** command to check the active version of the RPM.