

GwProvApp

- [Introduction](#)
- [Architecture](#)
- [Code Flow](#)

Introduction

The term "Provisioning" describes the ability to configure a Gateway with "information" that will be sent to its assets when those assets register with the Platform. The information is intended to help set up the asset for connections and communications with its managing Gateway.

GwProvApp (Gateway Provisioning Application) is the entry point for RDK-B middle-ware. It creates the state machine and does all gateway initialisation (eg: DOCSIS initialisation). It initiates the registration of all services on boot-up. It is responsible for provisioning of LAN, DOCSIS etc.

GwProvApp supports provisioning and managing of:

1. IPv4 and IPv6 devices, which include: Cable modems compliant with DOCSIS 1.0, 1.1, 2.0 and 3.0. Only DOCSIS 3.0 supports IPv6 devices.
2. Variants of eSAFE (embedded Service/Application Functional Entities) devices

Gateway Provisioning facilitates support many technologies to provide provisioning services for your network. These technologies include: eRouter, DOCSIS High-Speed Data.

The Data Over Cable Service Interface Specification (DOCSIS) defines functionality in cable modems that are involved in high-speed data distribution over cable television system networks. Using this feature, Multiple Systems Operators (MSOs) can provide a range of services (like broadband Internet connectivity, telephony) through Internet connection.

The primary function of the eRouter device is to allow subscribers to connect multiple CPE devices to the operator provided DOCSIS high-speed Internet service. DOCSIS specifications allow subscribers to directly connect multiple CPE devices to the cable modem; however, that requires operators to provide IP provisioning to each of the CPE devices. Depending on which IP Protocols are enabled, the eRouter allows provisioning of IPv4 CPEs, IPv6 CPEs, or Dual Stack (IPv4 and IPv6) CPEs simultaneously.

Architecture

The GW Provisioning Abstraction defines the abstracted APIs which are called by the gateway provisioning application during initialisation. It provides interfaces to be used for handshaking during DOCSIS initialisation.

gw_prov_abstraction.h file defines the abstracted APIs. gw_prov_utopia initialises and runs the provisioning process.

gw_prov_utopia create state machine for event handling from DOCSIS stack. This is called from provisioning abstraction layer when any provisioning event occurs. Registered DOCSIS init event will be handled by the state machine. gw_prov_utopia will initialise CCSP system configuration through utopia_init.sh.

The eRouter operates in any one of the following three modes:

- IPv4 Protocol Enabled
- IPv6 Protocol Enabled
- Dual IP Protocol Enabled

The eRouter can also be set to 'Disabled' mode, which turns the eRouter into a bridging device. The eRouter must support all the three modes of operations along with the ability to be set to 'Disabled' mode.

When eSAFE has not yet begun its provisioning process: Initialises DOCSIS resources required for external DOCSIS interface. All database resources needed for DOCSIS maintenance are initialised. Initialises an eSafe device database. Physical network interface used by the eRouter are added. DOCSIS events are registered.

At this point even if eRouter mode is enabled, eRouter operation mode is either "Disabled" or set to "No IPv4, no IPv6", regardless of which is enabled. DOCSIS local bridge is disconnected. Event notification is sent on DOCSIS initialisation.

When eSAFE is in the process of provisioning: When notification is received to change device mode to eRouter Mode (from disable to enable), enable eSafe device. Set the eRouter operation mode to "No IPv4, no IPv6".

eSAFE completes its provisioning:

When upstream ipv4-status notification is received, the eRouter operation mode is set to both IPv4, IPv6 or only IPv4 for IPv4 routing. When upstream ipv6-status notification is received, the eRouter operation mode is set to both IPv4, IPv6 or only IPv6 for IPv6 routing. This state is updated in eSafe device database to indicate that the eSAFE completed its provisioning process.

Code Flow

- GW provisioning is the first RDKB process to be invoked during boot-up after DOCSIS initialization
- As each GW is provisioned during boot up, GW provisioning application will be executed during every boot up

- GW provisioning is a generic provisioning module which provides necessary interfaces for OEMs to initialize DOCSIS give necessary callbacks based on the state
- Registers and listens to RDKB events and takes necessary action
- Initialized through PCD/systemd daemons depending on platform
- GW provisioning app initializes the provisioning DB and the necessary path configurations needed
- Provides file pointers for callback functions to OEMs which can be called based on certain events
- Calls utopia_init.sh script which initializes RDKB DBs and other services
- Takes action on TLVs coming through boot config file
- Handles creation and configuration of WAN interface
- Monitors link status and RDKB events in background

