QUICK START GUIDE

RUCKUS SmartZone (ST-GA) Getting Started on SZ GPB/MQTT Interface, 7.0.0

Supporting SmartZone 7.0.0

Part Number: 800-73393-001 Rev A
Publication Date: February 2024

Export Restrictions

These products and associated technical data (in print or electronic form) may be subject to export control laws of the United States of America. It is your responsibility to determine the applicable regulations and to comply with them. The following notice is applicable for all products or technology subject to export control:

These items are controlled by the U.S. Government and authorized for export only to the country of ultimate destination for use by the ultimate consignee or end-user(s) herein identified. They may not be resold, transferred, or otherwise disposed of, to any other country or to any person other than the authorized ultimate consignee or end-user(s), either in their original form or after being incorporated into other items, without first obtaining approval from the U.S. government or as otherwise authorized by U.S. law and regulations.

Disclaimer

THIS CONTENT AND ASSOCIATED PRODUCTS OR SERVICES ("MATERIALS"), ARE PROVIDED "AS IS" AND WITHOUT WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED. TO THE FULLEST EXTENT PERMISSIBLE PURSUANT TO APPLICABLE LAW, COMMSCOPE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TITLE, NON-INFRINGEMENT, FREEDOM FROM COMPUTER VIRUS, AND WARRANTIES ARISING FROM COURSE OF DEALING OR COURSE OF PERFORMANCE. CommScope does not represent or warrant that the functions described or contained in the Materials will be uninterrupted or error-free, that defects will be corrected, or are free of viruses or other harmful components. CommScope does not make any warranties or representations regarding the use of the Materials in terms of their completeness, correctness, accuracy, adequacy, usefulness, timeliness, reliability or otherwise. As a condition of your use of the Materials, you warrant to CommScope that you will not make use thereof for any purpose that is unlawful or prohibited by their associated terms of use.

Limitation of Liability

IN NO EVENT SHALL COMMSCOPE, COMMSCOPE AFFILIATES, OR THEIR OFFICERS, DIRECTORS, EMPLOYEES, AGENTS, SUPPLIERS, LICENSORS AND THIRD PARTY PARTNERS, BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER, EVEN IF COMMSCOPE HAS BEEN PREVIOUSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, WHETHER IN AN ACTION UNDER CONTRACT, TORT, OR ANY OTHER THEORY ARISING FROM YOUR ACCESS TO, OR USE OF, THE MATERIALS. Because some jurisdictions do not allow limitations on how long an implied warranty lasts, or the exclusion or limitation of liability for consequential or incidental damages, some of the above limitations may not apply to you.

Trademarks

CommScope and the CommScope logo are registered trademarks of CommScope and/or its affiliates in the U.S. and other countries. For additional trademark information see https://www.commscope.com/trademarks. All product names, trademarks, and registered trademarks are the property of their respective owners.

Patent Marking Notice

For applicable patents, see www.cs-pat.com.

Contents

Contact Information, Resources, and Conventions

Contacting RUCKUS Customer Services and Support

The Customer Services and Support (CSS) organization is available to provide assistance to customers with active warranties on their RUCKUS products, and customers and partners with active support contracts.

For product support information and details on contacting the Support Team, go directly to the RUCKUS Support Portal using https://support.ruckuswireless.com, or go to https://www.ruckusnetworks.com and select **Support**.

What Support Do I Need?

Technical issues are usually described in terms of priority (or severity). To determine if you need to call and open a case or access the self-service resources, use the following criteria:

- Priority 1 (P1)—Critical. Network or service is down and business is impacted. No known workaround. Go to the **Open a Case** section.
- Priority 2 (P2)—High. Network or service is impacted, but not down. Business impact may be high. Workaround may be available. Go to the **Open a Case** section.
- Priority 3 (P3)—Medium. Network or service is moderately impacted, but most business remains functional. Go to the **Self-Service Resources** section.
- Priority 4 (P4)—Low. Requests for information, product documentation, or product enhancements. Go to the **Self-Service Resources** section.

Open a Case

When your entire network is down (P1), or severely impacted (P2), call the appropriate telephone number listed below to get help:

- Continental United States: 1-855-782-5871
- Canada: 1-855-782-5871
- Europe, Middle East, Africa, Central and South America, and Asia Pacific, toll-free numbers are available at https://support.ruckuswireless.com/contact-us and Live Chat is also available.
- Worldwide toll number for our support organization. Phone charges will apply: +1-650-265-0903

We suggest that you keep a physical note of the appropriate support number in case you have an entire network outage.

Self-Service Resources

The RUCKUS Support Portal at https://support.ruckuswireless.com offers a number of tools to help you to research and resolve problems with your RUCKUS products, including:

- Technical Documentation—https://support.ruckuswireless.com/documents
- Community Forums—https://community.ruckuswireless.com
- Knowledge Base Articles—https://support.ruckuswireless.com/answers
- Software Downloads and Release Notes—https://support.ruckuswireless.com/#products_grid
- Security Bulletins—https://support.ruckuswireless.com/security

Using these resources will help you to resolve some issues, and will provide TAC with additional data from your troubleshooting analysis if you still require assistance through a support case or RMA. If you still require help, open and manage your case at https://support.ruckuswireless.com/case_management.

Document Feedback

RUCKUS is interested in improving its documentation and welcomes your comments and suggestions.

You can email your comments to RUCKUS at #Ruckus-Docs@commscope.com.

When contacting us, include the following information:

- Document title and release number
- Document part number (on the cover page)
- Page number (if appropriate)

For example:

- RUCKUS SmartZone Upgrade Guide, Release 5.0
- Part number: 800-71850-001 Rev A
- Page 7

RUCKUS Product Documentation Resources

Visit the RUCKUS website to locate related documentation for your product and additional RUCKUS resources.

Release Notes and other user documentation are available at https://support.ruckuswireless.com/documents. You can locate the documentation by product or perform a text search. Access to Release Notes requires an active support contract and a RUCKUS Support Portal user account. Other technical documentation content is available without logging in to the RUCKUS Support Portal.

White papers, data sheets, and other product documentation are available at https://www.ruckusnetworks.com.

Online Training Resources

To access a variety of online RUCKUS training modules, including free introductory courses to wireless networking essentials, site surveys, and products, visit the RUCKUS Training Portal at https://commscopeuniversity.myabsorb.com/. The registration is a two-step process described in this video. You create a CommScope account and then register for, and request access for, CommScope University.

Document Conventions

The following table lists the text conventions that are used throughout this guide.

TABLE 1 Text Conventions

Convention	Description	Example
`monospace`	Identifies command syntax examples	`device(config)# interface ethernet 1/1/6`
bold	User interface (UI) components such as screen or page names, keyboard keys, software buttons, and field names	On the **Start** menu, click **All Programs**.
italics	Publication titles	Refer to the *RUCKUS Small Cell Release Notes* for more information.

Notes, Cautions, and Safety Warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A NOTE provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An ATTENTION statement indicates some information that you must read before continuing with the current action or task.

CAUTION
A CAUTION statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A DANGER statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Command Syntax Conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{**x**| **y**| **z**}	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x|**y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a “soft” line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

New In This Document

TABLE 2 Key Features and Enhancements in *SmartZone 7.0.0 Rev A* (*February 2024*)

Feature	Description	Reference
FlowMessage	**Updated**: App_Id	ap_avc.proto on page 24
APClientInfo	**Updated**: • Client capabilities of Multi Link Operation like EMLSR or EMLMR • Client link of Multi Link Operation • Client MLD Mac address • MLO Client Active link • Wlan auth status • Wlan authentication method • Encryption method used by the AP • Reason for disconnect from the controller.	ap_client.proto on page 31
APPeer	**Updated**: 11mc measurement distance in meters	ap_peerlist.proto on page 44
APPeerReport	**Updated**: • Ture if the ap is rebooted and exceed a reboot threshold, which is the duration from the previous shutdown time to the boot up time. • AP's serial number	
AP Report Bin WLAN Information	**Updated**: • Probes dropped due to RSSI/SNR/PPS/ATD/TCM Threshold features • duth reqs dropped due to RSSI/SNR/PPS/ATD/TCM Threshold features • 11v BTM requests sent by AP for 11v Steering features (CLB, BB, SROAM) • STA Kicked due to Smart Roam Algorithm • Probes dropped due to Smart Roam Algorithm • Auth requests dropped due to Smart Roam Algorithm	ap_report.proto on page 46

TABLE 2 Key Features and Enhancements in *SmartZone 7.0.0 Rev A* (*February 2024*) (continued)

Feature	Description	Reference
APStatusWlan	**Updated**: • UE session lookup success from fst • UE session lookup success from sessionMgr • roaming success • Number of connection attempts • Number of successful PMKID creation • Number of the PMKID in client's reassoc not found in the AP pmksa • Number of the PMKID in the re-assoc is not presented to the AP • Number of FT roam has successfully completed • Number of parameters mismatched in FTIE • Number of parameters missing in FTIE	ap_status.proto on page 67
APStatusData	**Updated**: Serialization data for AFC information	ap_status.proto on page 67
APStatusSystem	**Updated**: Secure boot status (0: disabled 1: enabled)	ap_status.proto on page 67
MockSCI-TLS	**Updated**: SZ to SCI MQTT subscriber software for CentOS / Ubuntu.	Executing the Test Subscriber on page 17
GPB.proto image	**Updated**: Google ProtoBuf image for GPB/MQTT.	Compiling Google Protobuf Binding Classes on page 16

GPB-MQTT Interface Implementation

GPB-MQTT Overview

The controller GPB-MQTT is an interface that allows an external application to receive the statistical data from an Access Point (AP) managed by the controller.

The statistical data includes device information, event records, access point statistics, client statistics, wireless radio or network statistics and rogue AP data. The streaming data is presented in Google Protocol Buffer (GPB) format. The external application can use the program library compiled with GPB data structure to read the data.

Message Queuing Telemetry Transport (MQTT) is a simple, lightweight, and publish-subscribe messaging protocol. It is useful for connections with remote locations where a small code footprint is required or the network bandwidth is at a premium.

Prerequisite Task

Before implementing the GPB/MQTT interface, some background infrastructure must be installed and configured.

NOTE
Installation steps are provided for CentOS-6 and Ubuntu users. If you have different platform requirements, refer to the Mosquitto website for other installation guides: https://www.mosquitto.org

1. Use a Unix-compatible operating system (for example: DebianWheezy, DebianJessie, CentOS 6.6, Mac OSX 10.10, Mac OSX 10.11)
2. Download and install a compatible JDK version 1.8 (OpenJDK8, OracleJDK (Java SE 8u161/8u162)). You can also use JDK8.

   If you are using a Linux OS, use OpenJDK 8 and download it from http://openjdk.java.net/install/. If you are working on Mac OX or Windows, refer to the Oracle website for downloading the latest JDK 8 http://www.oracle.com/technetwork/java/javase/overview/index.html.
3. Download and install Gradle version 2.9.x from https://gradle.org/releases/.
4. Download and install Mosquitto 1.4.x MQTT broker, MQTT client, and requisite client libraries from http://mosquitto.org/download/.

5. If you are running CentOS-6, follow these steps:
   a) Run the following command:

```
$ cd /etc/yum/yum.repos.d
```

   b) Add the following content into filehome-oojah-mqtt.repo.

```
[home_oojah_mqtt]
name=mqtt (CentOS_CentOS-6) type=rpm-md
baseurl=http://download.opensuse.org/repositories/home:/oojah:/mqtt/ CentOS_CentOS-6/
gpgcheck=1 gpgkey=http://download.opensuse.org/repositories/home:/oojah:/mqtt/C entOS_CentOS-6//
repodata/repomd.xml.key
enabled=1
```

   c) Run the following command:

```
$ sudo yum update
```

   d) Run the following command:

```
$ sudo yum install mosquito mosquito-clients
```

6. If you are running Ubuntu, install Mosquitto MQTT broker with the following commands:

```
$sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa
$sudo apt-get update
$sudo apt-get install mosquitto mosquitto-clients
```

7. Open the firewall to ports 1883 and 8883.
8. Start the mosquito broker on an instance which can be reached from the instance where the sample client is executed.
9. Configure the controller with the Mosquitto IP address and port to transfer SCI data.

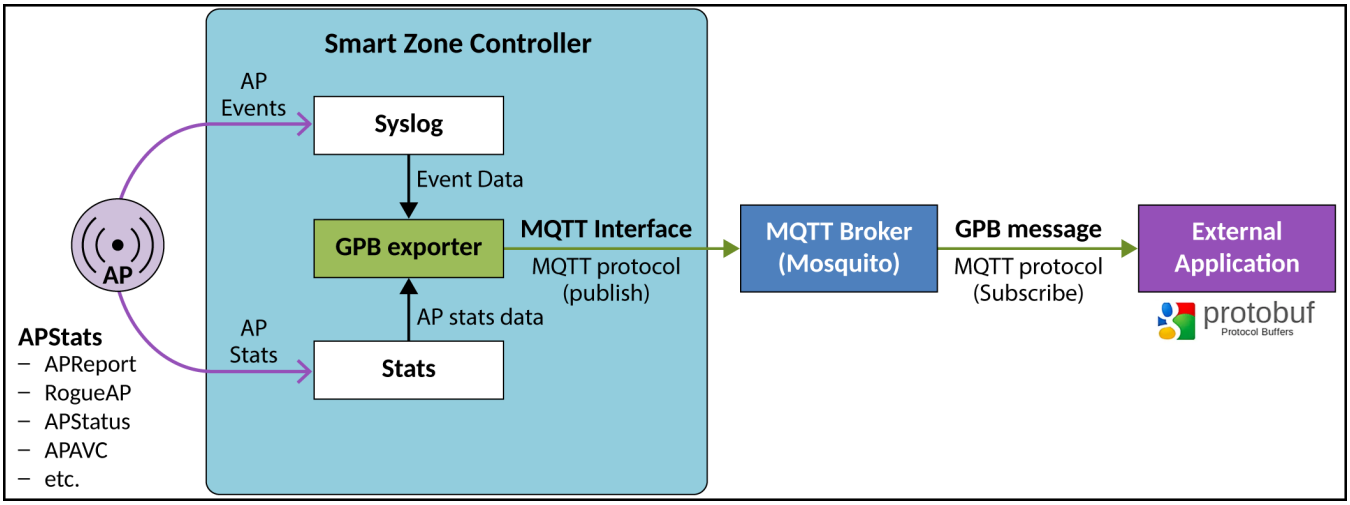
   Refer to the SmartZone technical documentation to achieve this.
10. Enable SCI in the MQTT broker.

Working with the GPB-MQTT Interface

To work with the GPB-MQTT interface use the following steps.

1. Set the MQTT broker to receive GPB data.
2. Configure the MQTT broker IP address on the controller web user interface to publish to GPB data to MQTT broker.
3. Use the GPB compiler to get the library for reading the data with SZ data structure (.proto files). The SmartZone (SZ) GPB .proto files can be downloaded from the RUCKUS support site at: https://support.ruckuswireless.com
4. The external application implements the MQTT protocol and subscribes the topic to MQTT broker.
5. The external application receives the GPB data from MQTT broker and then uses the GPB library to read the streaming data.
6. The external application converts the GPB data and saves it to its local system.

FIGURE 1 GPB-MQTT Interface Diagram



Enabling Authentication in the MQTT Broker

The current implementation of the controller needs authentication to the MQTT Broker.

Perform the following steps to create a profile used in the authentication process with MQTT.

NOTE
RUCKUS recommends that you do not start the Mosquitto MQTT broker by user root. If you start Mosquitto MQTT broker by user root or with sudo, it will result in an incorrect start-up environment or other error.

1. Startup MQTT broker with security enabled.
   a) Create a mosquitto configuration file, mosquitto.conf. Refer to the examples below.

```
listener 8883
psk_file /etc/mosquitto/pskfile
psk_hint hint
tls_version tlsv1.2
```

   b) According to the pskfile configuration of mosquitto.conf, you must create another pskfile to store the authentication pre-shared key. The format is <*username*>:<*password phrase*>. The password phrase is converted into hex representation. Here is an example for the pskfile:

```
testuser:7465737475736572
```

   The original password text before converting is **testuser**. The password has to match with the setting configured on the controller web interface.

   c) After you install the mosquitto MQTT broker, you can start it up by the following command:

```
mosquitto -c /home/user/mosquitto.conf -p 1883
```

NOTE
You can terminate the MQTT broker by pressing **ctrl + C** on the MQTT broker console.

2. After starting an MQTT subscriber, you can start another MQTT subscriber to connect to the MQTT broker using unauthenticated port 1883 or authenticated port 8883.

   a) Here is an example to connect to the MQTT broker with unauthenticated version:

   ```
   mosquitto_sub -h <mosquitto_ip> -p 1883 -t 'sci-topic'
   ```

   b) Here is an example to connect to the MQTT broker using an authenticated security channel:

   ```
   mosquitto_sub -h <mosquitto_ip> -p 8883 -t 'sci-topic' --psk-identity testuser --psk
   7465737475736572
   ```

   You can find the difference between the unauthenticated and authenticated version of the mosquitto_sub function, where the psk-identity and psk attribute is connected to MQTT broker using the port number 8883.

3. Use the following information to create or configure the MQTT connection profile :

   a) MQTT server name.

   b) MQTT server host / IP address.

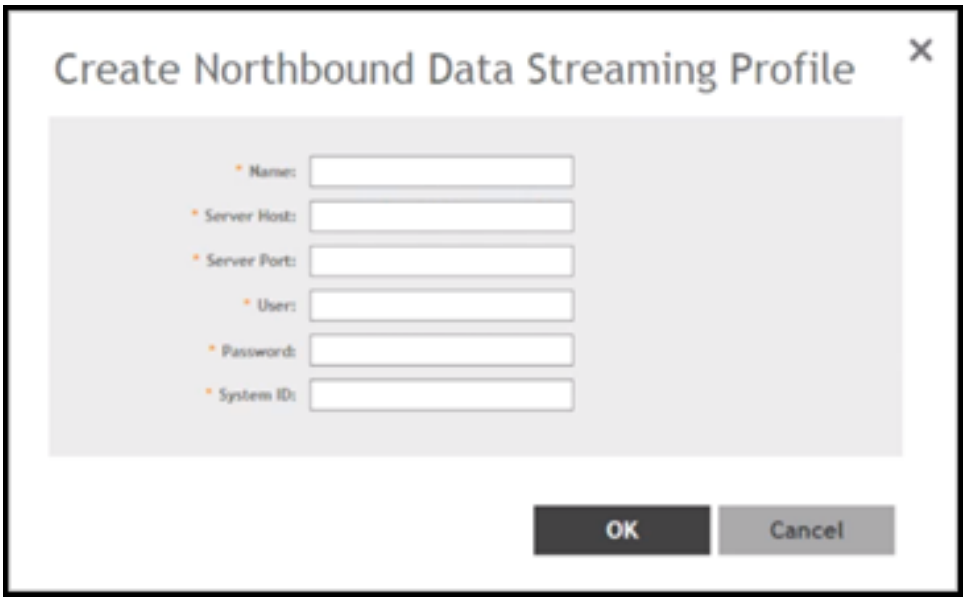
   c) MQTT server port number.

   d) System ID—The backend system collects data from multiple controllers. The system identifier is used to distinguish the data source.

   e) User credentials of login name and password.

   In the authenticated mode of SCI profile, you must provide the user name and password for TLSv1.2 with pre-shared key exchange standard as shown in the following figure.

   FIGURE 2 Setting SCI Profile in Authenticated Mode



   The SCI profile setting is used for allowing the controller to get the MQTT connection information. For example: IP address, port, or user credential. It does not mean that you must have a SCI setup to receive the GPB streaming data.

4. Save the SCI profile.

5. Configure the Northbound data streaming as explained in the following section.

Configuring Northbound Data Streaming Settings

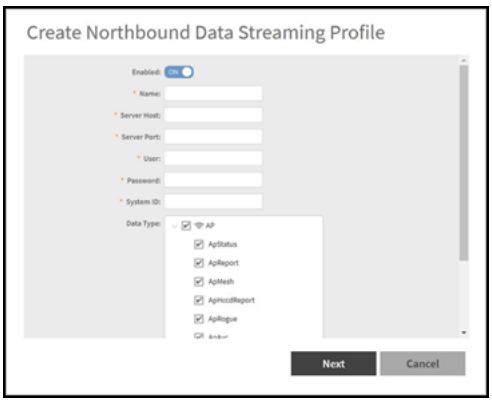
Configuring the Northbound Data Streaming settings in the controller enables data transfer from the controller to the Northbound Data Streaming server using the Message Queuing Telemetry Transport (MQTT) protocol.

NOTE
You can create a maximum of two SCI profiles simultaneously.

Complete the following steps to configure the Northbound Data Streaming server settings.

1. From the main menu, go to **Administrator** > **External Services** > **Northbound Data Streaming**.
2. Click **Create**. The **Create Northbound Data Streaming Profile** dialog box is displayed.

FIGURE 3 Creating a Northbound Data Streaming Profile



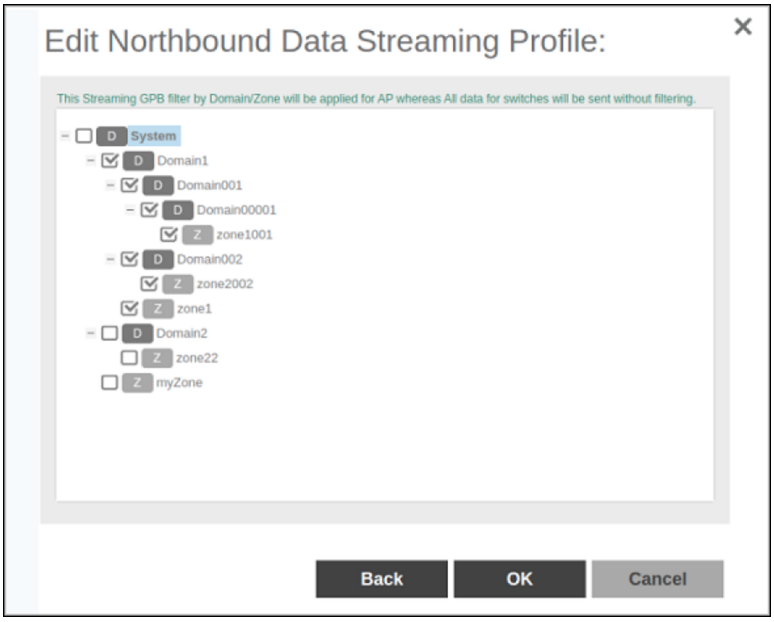
3. Complete the following options:
   - **Enabled**: Set to **ON** to configure the Northbound Data Streaming profile.
   - **Name**: Enter the profile name.
   - **Server Host**: Enter the IP address of the Northbound Data Streaming host server.

     NOTE
     An SCI profile supports only the IPv4 format.

   - **Server Port**: Enter the port number using which the Northbound Data Streaming server and the controller can communicate and transfer data. The ports must be allowed on the firewall.
   - **User**: Enter the name of the user.
   - **Password**: Enter the password for the user.
   - **System ID**: Enter the ID of the Northbound Data Streaming system to access.
   - **Data Type**: Select the required options for the specific data types that must be sent to the Northbound Data Streaming server from the SCI server.
4. Click **Next**.

5. For APs, from the **System** tree, select the required domain or zone to send KPIs or statistics to the Northbound Data Streaming server. For switches, KPIs or the statistics are sent to SCI or Northbound Data Streaming server without filtering.

   FIGURE 4 Selecting the Zone or Domain



6. Click **OK**.

   The Northbound Data Streaming profile is listed on the Northbound Data Streaming page.

   The **Status** column displays the current connection status of the SCI profile.

   NOTE
   You can also edit or delete a Northbound Data Streaming profile by selecting the Northbound Data Streaming profile and clicking the **Configure** or **Delete** option.

Compiling Google Protobuf Binding Classes

Protocol Buffers, referred to as Protobuf, is widely used at Google for storing and interchanging all kinds of structured information.

Perform the following steps to compile the Google Protobuf (GPB) binding class.

1. Download the latest controller GPB .proto files from the RUCKUS support site at: https://support.ruckuswireless.com/software/3962-smartzone-7-0-0-ga-gpb-proto-google-protobuf-image-for-gpb-mqtt.
2. Follow the compiling instructions for getting the binding classes for different languages. For more information, refer to https://developers.google.com/protocol-buffers/.
3. If you have not installed the compiler, download the protoc compiler version 2.6.1, and follow the instructions in the README file.

4. Run the compiler and specify the following:
   - Source directory where the source code of your application is stored. The current directory is used if you do not provide a value.
   - Destination directory where you want the generated code to be stored. This is usually the same as SRC_DIR.
   - Path to the .proto file.

   We can use **ap_client.proto** as an example for this task. You can substitute any **.proto** file.
5. Generate the classes for reading and writing the AP Client GPB messages. To do this, you need to run the protocol buffer compiler protoc on your .proto file by:
   a) Java—Run the following script using the *java_out* option for Java classes. Similar options are provided for other supported languages .

      ```
      protoc -I=$SRC_DIR --java_out=$DST_DIR $SRC_DIR/ap_client.proto
      ```

      This generates *com/ruckuswireless/scg/protobuf/APClient.java* in your specified destination directory.
   b) Python—Run the following script using the *python_out* option for Python classes. Similar options are provided for other supported languages.

      ```
      protoc -I=$SRC_DIR --python_out=$DST_DIR $SRC_DIR/ap_client.proto
      ```

      This generates *apclient_pb2.py* in your specified destination directory.
   c) C++—Run the following script using the*cpp_out* option for C++ classes. Similar options are provided for other supported languages .

      ```
      protoc -I=$SRC_DIR --cpp_out=$DST_DIR $SRC_DIR/ap_client.proto
      ```

      This generates the following files in your specified destination directory:
      - apclient.pb.h—The header which declares your generated classes.
      - apclient.pb.cc—Contains the implementation of your classes.
6. Copy the requisite technology stack GPB binding classes or source files to your project, which is used for receiving the controller's GPB streaming data. It can be used to decode and parse the content of GPB message data.

Executing the Test Subscriber

The test subscriber is a utility provided by RUCKUS to receive the GPB streaming data from the controller.

The test subscriber utility is written in Java. It uses the Java classes, which is compiled with RUCKUS GPB .proto to read the content of GPB message.

> **NOTE**
> The test subscriber utility is designed only for test purposes and not for integrating with your application of the controller GPB-MQTT interface.

Downloading the Subscriber Software

Download the subscriber software from: https://support.ruckuswireless.com/software/3963-smartzone-7-0-0-ga-mocksci-tls-sz-to-sci-mqtt-subscriber-software-for-centos-ubuntu-dnp.

Execution Script to Start Mosquitto MQTT

Execute the following command to start the Mosquitto MQTT broker .

```
mosquitto -c /etc/mosquitto/mosquitto.conf -p 1883 &
```

Execution Script for Subscriber Software Version Upto 3.6.x

Execute the following script to write all received messages to the SciTlsMessages folder in the program execution directory.

```
/execute-normal.sh <mqtt broker IP address> <mqtt port number of security channel> <s/n>
```

The option s/n refers to:

- s—Scaling mode that shows the statistical counter result.
- n— Normal mode that writes to the receiving file from the MQTT broker.

The following example shows where the script connects to a MQTT broker on 172.17.18.144:8883 in scaling mode.

```
./execute-normal.sh 172.17.18.144 8883 s
```

Execution Script for Subscriber Software Version 5.0.x and Above

Start the Mock SCI using the below scripts for different types of topic service subscriber.

- For SCI topic service, execute the below startup command with the arguments of MQTT broker IP address, port number, and to enable (true) or disable (false) the scaling mode.

```
# Execute in Normal Mode
./execute-sci.sh 172.17.18.144 8883 false

# Execute in Scaling Mode
./execute-sci.sh 172.17.18.144 8883 true
```

- For GStation topic service, execute the below startup command with the arguments of MQTT broker IP address, port number, and to enable (true) or disable (false) the scaling mode.

```
# Execute in Normal Mode
./execute-gstation.sh 172.17.18.144 8883 false

# Execute in Scaling Mode
./execute-gstation.sh 172.17.18.144 8883 true
```

Execution Script for Subscriber Software Version 6.x and Above

1. Open the .tar file. For example, `zxvf scg-mock-sci-6.1.0-20211129.082645-79.tar.gz`

```
□ □ /tmp/b/scg-mock-sci-6.1.0-SNAPSHOT_20211129082535 □ tree ./
./
├── run_localhost.sh
├── run.sh
└── scg-mock-sci-6.1.0-SNAPSHOT.jar

0 directories, 3 files
```

2. Execute Mock SCI.
   - If mosquitto is setup on the same device, execute the following command:

```
□ □ /tmp/b/scg-mock-sci-6.1.0-SNAPSHOT_20211129082535 □ ./run_localhost.sh
```

   a. Choose topic to subscribe.

```
Please choose topic to subscribe:
1.sci-topic
2.gstation-topic
Please enter 1 or 2: 1
Topic selected is: sci-topic
hostname/IP entered is: localhost
port entered is: 8883
```

b. Enable or disable scaling mode (**enable** : show subscribed data statistics on StdOut rather than writing detailed data to file system).

```
Please enter yes/no to enable/disable scaling mode: no
```

- If mosquitto is setup on a different device, execute the followin command:

```
✗ □ □ /tmp/b/scg-mock-sci-6.1.0-SNAPSHOT_20211129082535 □ ./run.sh
```

a. Choose topic to subscribe.

```
Please choose topic to subscribe:
1.sci-topic
2.gstation-topic
Please enter 1 or 2: 1
Topic selected is: sci-topic
```

b. Specify hostname or IP address to connect to the remote Mosquitto service.

```
Please enter hostname or IP (characters length < 100): 10.206.79.133
hostname/IP entered is: 10.206.79.133
```

c. Specify port to connect remote Mosquitto service.

```
Please enter port between 1000 and 65535: 8883
port entered is: 8883
```

d. Enable or disable scaling mode (**enable** : show subscribed data statistics on StdOut rather than writing detailed data to filesystem).

```
Please enter yes/no to enable/disable scaling mode: no
```

Exit from the Test subscriber

Type **EXIT** and click on the **Enter** key to leave the test subscriber.

Execution Result

All receiving GPB messages from MQTT's publisher (example: controller) will be saved to the SciTlsMessages folder.

Appendix

AP Message Hierarchy and Information

The following diagrams indicate the GPB status hierarchy, along with each protocol file and field descriptions.

FIGURE 5 GPB Message Hierarchy

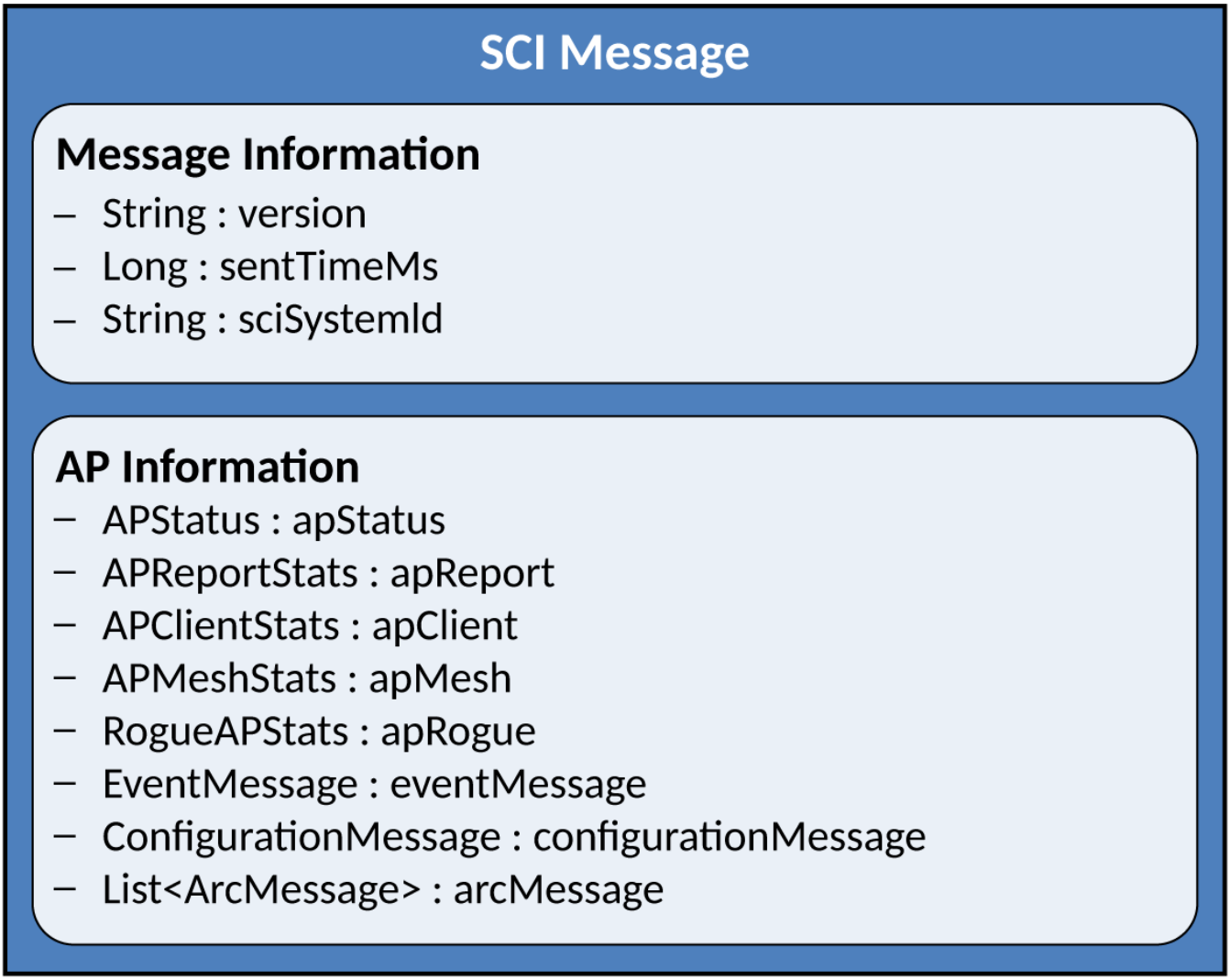
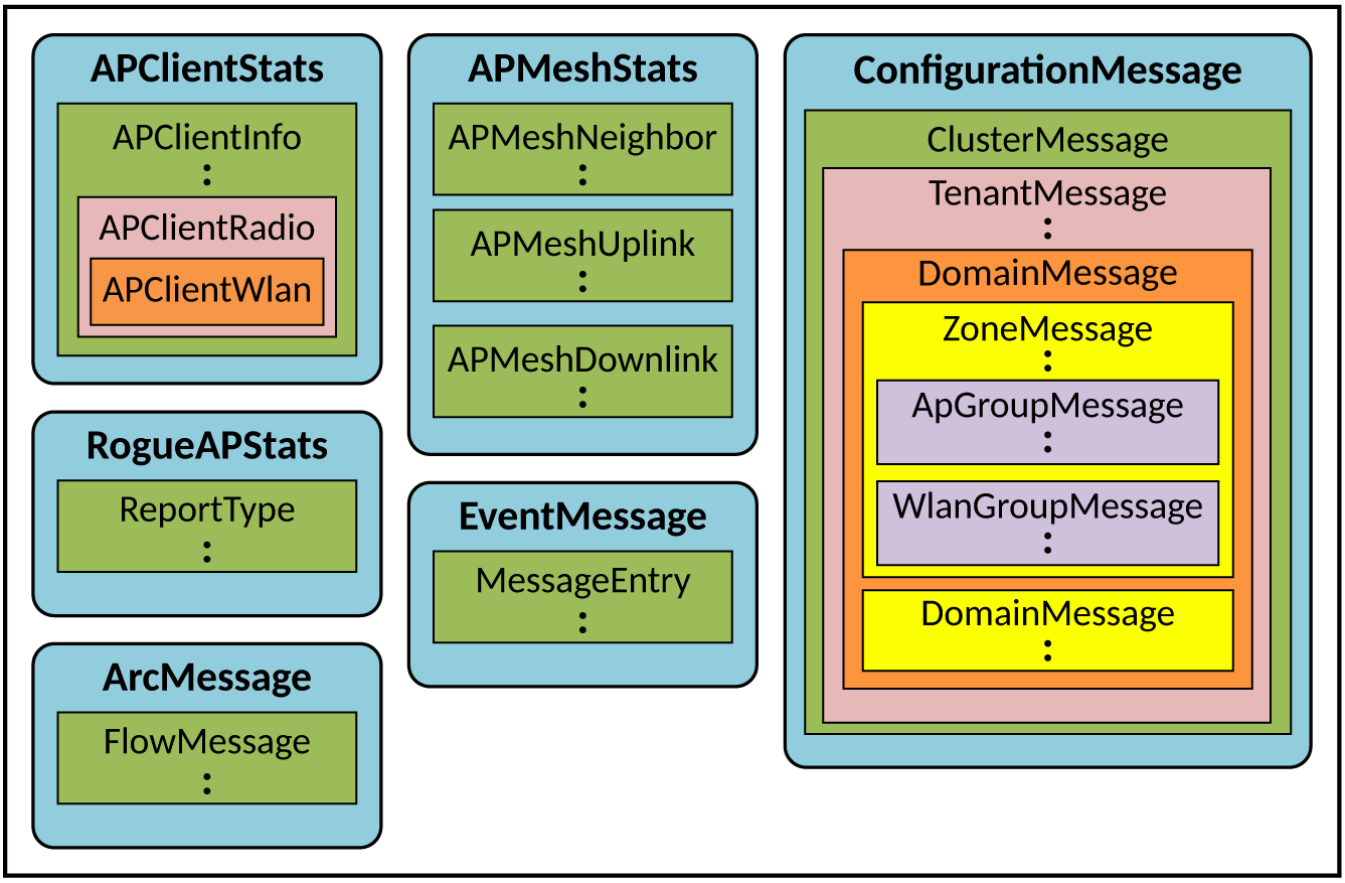


FIGURE 6 AP Information



ap_avc.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message FlowMessage {
    required string app = 1;
    required uint32 port = 2;
    required string client_mac = 3;
    required string ap_mac = 4;
    required string ssid = 5;
    required uint64 uplink = 6;
    required uint64 downlink = 7;
    required uint64 total = 8;
    optional string category = 9;
    optional string wlangroup_id = 10;
    optional uint32 wsgwlan_id = 11;
    optional string wlantenant_id = 12;
    optional uint32 apradiotype_id = 13;
    optional string wlangroup_name = 14;
    optional string wlantenant_name = 15;
    optional uint32 radio_id = 16;
    optional string client_ipv4 = 17;
    optional string client_ipv6 = 18;
    optional string client_hostname = 19;
    required uint32 app_id = 20;
}

message UrlFilteringMsg {
    enum Action {
        Allow = 1;
        Block = 2;
    }
    optional string url = 1;
    optional uint32 cat_id = 2;
    optional string cat_name = 3;
    optional uint32 num_hits = 4;
    optional string client_mac = 5;
    optional string ap_mac = 6;
    optional string ssid = 7;
    optional uint64 uplink = 8;
    optional uint64 downlink = 9;
    optional uint64 total = 10;
    optional string wlangroup_id = 11;
    optional uint32 wsgwlan_id = 12;
    optional string wlantenant_id = 13;
    optional uint32 apradiotype_id = 14;
    optional string wlangroup_name = 15;
    optional string wlantenant_name = 16;
    optional uint32 radio_id = 17;
    optional string client_ipv4 = 18;
    optional string client_ipv6 = 19;
    optional string client_hostname = 20;
    optional uint64 session_start = 21;
    optional uint64 session_stop = 22;
    optional uint32 web_reputation_score = 23;
    optional Action web_reputation_action = 24;
}

message WFCCallQualityMsg {
    optional uint64 timestamp = 1;
    optional uint32 score = 2;
}
```

```
message WifiCallingMsg {
    optional string operator_name = 1;
    optional uint32 priority = 2;
    optional uint64 timestamp_start = 3;
    optional uint64 timestamp_end = 4;
    optional string client_mac = 5;
    optional string ap_mac = 6;
    optional string ssid = 7;
    optional uint64 uplink = 8;
    optional uint64 downlink = 9;
    optional uint64 total = 10;
    optional string wlangroup_id = 11;
    optional uint32 wsgwlan_id = 12;
    optional string wlantenant_id = 13;
    optional uint32 apradiotype_id = 14;
    optional string wlangroup_name = 15;
    optional string wlantenant_name = 16;
    optional uint32 radio_id = 17;
    optional string client_ipv4 = 18;
    optional string client_ipv6 = 19;
    optional string client_hostname = 20;
    optional string epdg_fqdn = 21;
    repeated WFCCallQualityMsg wfc_score_dir1 = 22;
    repeated WFCCallQualityMsg wfc_score_dir2 = 23;
}

message ServerInfo {
    required string server_ip = 1;
    optional float lan_latency = 2;
    optional float wan_latency = 3;
    optional float page_download_time = 4;
}

message ApplicationData {
    required uint32 app_id = 1;
    repeated ServerInfo server_info = 2 [(nanopb).type = FT_POINTER];
}

message ClientInfo {
    required string client_ip = 1;
    optional string client_mac = 2;
    repeated ApplicationData app_data = 3 [(nanopb).type = FT_POINTER];
}

message ApplicationMetrics {
    repeated ClientInfo client_info = 1 [(nanopb).type = FT_POINTER];
}

message ArcMessage {
    optional string ver = 1;
    optional string zone_id = 2;
    optional string apgroup_id = 3;
    repeated FlowMessage rep_flow = 4;
    optional uint64 timestamp = 5;
    optional string cluster_id = 6;
    optional string domain_id = 7;
    optional string aptenant_id = 8;
    optional string map_id = 9;
    optional string aptenant_name = 10;
    optional string zone_name = 11;
    optional string apgroup_name = 12;
    optional string domain_name = 13;
    optional uint64 sampleTime = 14;
    optional uint32 aggregationInterval = 15;
    optional string apMac = 16
    repeated UrlFilteringMsg url_info = 17;
    repeated WifiCallingMsg wfc_info = 18;
    optional string apRadioDeploy = 19;
    optional ApplicationMetrics app_metrics = 20;

}
```

Field Description

TABLE 3 FlowMessage descriptions

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
app	string	snapshot	NULL	Application name
port	uint32	snapshot	NULL	Application destination port number
client_mac	string	snapshot	NULL	Destination MAC address
ap_mac	string	snapshot	NULL	AP MAC address
ssid	string	snapshot	NULL	WLAN SSID
uplink	uint64	snapshot	SUM	Rx bytes for this application
downlink	uint64	snapshot	SUM	Tx bytes for this application
total	uint64	snapshot	SUM	Tx + Rx bytes for the application
category	string	snapshot	NULL	Application category
wlangroup_id	string	snapshot	NULL	WLAN group identifier
wsgwlan_id	uint32	snapshot	NULL	WSG WLAN identifier
apradiotype_id	uint32	snapshot	NULL	Radio mode for the radio interface
wlangroup_name	string	snapshot	NULL	WLAN group name
wlantenant_name	string	snapshot	NULL	WLAN tenant name
radio_id	uint32	snapshot	NULL	WLAN radio ID
client_ipv4	string	snapshot	NULL	Client's IPv4 address
client_ipv6	string	snapshot	NULL	Client's IPv6 address
client_hostname	string	snapshot	NULL	Client host name
app_id	uint32	snapshot	NULL	App_Id

TABLE 4 UrlFilteringMsg descriptions

Attribute Name	ValueType (size)	Description
Allow	1	Web Reputation action value Allow
Block	2	Web Reputation action value Block

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
url	string	snapshot	NULL	Client access URL
cat_id	uint32	snapshot	NULL	Category identifier of the URL
cat_name	string	snapshot	NULL	Category name of the URL
num_hits	uint32	snapshot	SUM	Number of client counts that access the URL
client_mac	string	snapshot	NULL	Client's MAC address
ap_mac	string	snapshot	NULL	AP MAC address
ssid	string	snapshot	NULL	WLAN SSID where client is connected
uplink	uint64	snapshot	SUM	Uplink bytes sent from client
downlink	uint64	snapshot	SUM	Downlink bytes received by client
total	uint64	snapshot	SUM	Total of sent and received bytes by the client

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
wlangroup_id	string	snapshot	NULL	WLAN group identifier
wsgwlan_id	uint32	snapshot	NULL	WSG WLAN identifier
wlantenant_id	uint32	snapshot	NULL	WLAN tenant identifier
apradiotype_id	uint32	snapshot	NULL	Radio mode for the radio interface
wlangroup_name	string	snapshot	NULL	WLAN group name
wlantenant_name	string	snapshot	NULL	WLAN tenant name
radio_id	uint32	snapshot	NULL	Radio interface identifier
client_ipv4	string	snapshot	NULL	Client's IPv4 address
client_ipv6	string	snapshot	NULL	Client's IPv6 address
client_hostname	string	snapshot	NULL	Client host name
session_start	uint64	snapshot	NULL	URL Filtering session start timestamp
session_stop	uint64	snapshot	NULL	URL Filtering session start timestamp
web_reputation_score	unit32	snapshot	NULL	URL's web reputation score
web_reputation_action	.UrlFilteringMsg.Actio n	snapshot	NULL	Web reputation Action of the URL

TABLE 5 WFCCallQualityMsg descriptions

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG, NULL)	Description
timestamp	uint64	None	None	None
score	uint32	None	None	None

TABLE 6 WifiCallingMsg descriptions

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
operator_name	string	snapshot	NULL	Operator name
priority	uint32	snapshot	NULL	Call priority
timestamp_start	uint64	snapshot	NULL	Date and time at the start of the WiFi call
timestamp_end	uint64	snapshot	NULL	Date and time at the end of the WiFi call
client_mac	string	snapshot	NULL	Client's MAC address
ap_mac	string	snapshot	NULL	AP MAC address
ssid	string	snapshot	NULL	WLAN SSID where client is connected
uplink	uint64	snapshot	SUM	Uplink bytes sent from client
downlink	uint64	snapshot	SUM	Downlink bytes received by client
total	uint64	snapshot	SUM	Total of sent and received bytes by the client
wlangroup_id	string	snapshot	NULL	WLAN group identifier
wsgwlan_id	uint32	snapshot	NULL	WSG WLAN identifier
wlantenant_id	uint32	snapshot	NULL	WLAN tenant identifier

TABLE 6 WifiCallingMsg descriptions (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
apradiotype_id	uint32	snapshot	NULL	AP Radio mode
wlangroup_name	string	snapshot	NULL	WLAN group name
wlantenant_name	string	snapshot	NULL	WLAN tenant name
radio_id	uint32	snapshot	NULL	Radio interface identifier
client_ipv4	string	snapshot	NULL	Client's IPv4 address
client_ipv6	string	snapshot	NULL	Client's IPv6 address
client_hostname	string	snapshot	NULL	Client host name
epdg_fqdn	string	snapshot	NULL	FQDN of operator epdg gateway
wfc_score_dir1	WFCCallQualityMsg	snapshot	NULL	Call Quality Score
wfc_score_dir2	WFCCallQualityMsg	None	None	None

TABLE 7 ServerInfo descriptions

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type (SUM,MAX,MIN,AVG,NULL)	Description
server_ip	STRING	snapshot	NULL	Server IP address
lan_latency	FLOAT	snapshot	NULL	LAN latency
wan_latency	FLOAT	snapshot	NULL	WAN latency
page_download_time	FLOAT	snapshot	NULL	Page Download Time

TABLE 8 ApplicationData descriptions

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type (SUM,MAX,MIN,AVG,NULL)	Description
app_id	UINT32	snapshot	NULL	Application identifier
server_info	.ServerInfo	snapshot	NULL	Server Information

TABLE 9 ClientInfo descriptions

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type (SUM,MAX,MIN,AVG,NULL)	Description
client_ip	STRING	snapshot	NULL	Client IP address
client_mac	STRING	snapshot	NULL	Client Mac address
app_data	.ApplicationData		snapshot	Application Data information

TABLE 10 ApplicationMetrics description

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type (SUM,MAX,MIN,AVG,NULL)	Description
client_info	.ClientInfo	snapshot	NULL	Client information contains App data

TABLE 11 ArcMessage descriptions

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
version	string	snapshot	NULL	Arc data version number
zone_ID	string	snapshot	NULL	Zone UUID
apgroup_ID	string	snapshot	NULL	AP group UUID
rep_flow	.FlowMessage	snapshot	NULL	ARC stats message type
timestamp	uint64	snapshot	NULL	Timestamp for the message reported
cluster_id	string	snapshot	NULL	Cluster identifier
domain_id	string	snapshot	NULL	Domain identifier
aptenant_id	string	snapshot	NULL	AP tenant identifier
map_id	string	snapshot	NULL	MAP identifier
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name
apgroup_name	string	snapshot	NULL	AP group name
domain_name	string	snapshot	NULL	Domain name
sampleTime	uint64	snapshot	NULL	AVC data sampling timestamp
aggregationInterval	uint32	snapshot	NULL	Stats aggregation interval
apMac	string	snapshot	NULL	AP MAC address
url_info	.UrlFilteringMsg	snapshot	NULL	URL Filtering Stats message type
wfc_info	.WifiCallingMsg	snapshot	NULL	WiFi Calling stats message type
apRadioDeploy	string	snapshot	NULL	Indicate AP all radio band deploy
app_metrics	.ApplicationMetrics	NULL	NULL	Application QOE score

ap_avc_all.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APAVCStats {
    required uint32 version = 1;
    required .ArcMessage arc_message = 2;
}
```

Field Description

TABLE 12 APAVCStats descriptions

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
version	uint32			
arc_message	.ArcMessage			

ap_client.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APClientInfo {
    required string clientMac = 1;
    optional string ipAddress = 2;
    optional string ipv6Address = 3;
    optional int32 wlanId = 4;
    optional int32 rssi = 5;
    optional int32 receiveSignalStrength = 6;
    optional int32 noiseFloor = 7;
    optional int32 vlan = 8;
    optional uint64 rxFrames = 9;
    optional uint64 rxBytes = 10;
    optional uint64 txFrames = 11;
    optional uint64 txBytes    = 12;
    optional uint64 txMgmtFrames = 13;
    optional uint64 rxMgmtFrames = 14;
    optional uint32 throughputEst = 15;
    optional uint64 txDropDataFrames = 16;
    optional uint64 txDropMgmtFrames = 17;
    optional uint32 rxCRCErrFrames = 18;
    optional uint32 txRetry = 19;
    optional string osType = 20;
    optional APClientRadio radio = 21;
        message TCWithQuota {
                optional string tcName = 1;
                optional string tcMaxQuota = 2;
                optional string tcRemainingQuota =3;
        }
 repeated TCWithQuota tcWithQuota = 22;
    optional string cpeMac = 23;
    optional uint32 stickyWeak = 24;
    optional int32 deviceType = 25;
    optional int32 osVendorType = 26;
    optional string modelName = 27;
    optional string hostname = 28;
    optional uint32 medianTxMCSRate = 29;
    optional uint32 medianRxMCSRate = 30;
    optional string default_gateway = 31;
    optional string dhcp_server_ip = 32;
    optional string subnet_mask = 33;
    optional string dns_server_list = 34;
    optional string mlo_capability = 35;
    optional string mlo_links = 36;
    optional string mld_addr = 37;
    optional bool active_link = 38;


    /* jump to index 1000, for new requirement from SNMP and SCI */
    optional string ConnectMode = 1001;
    optional string Username = 1002;
    optional string SessionId = 1003;
    optional string MultipleSessionId = 1004;
    optional string AuthMode = 1005;
    optional uint64 DiscTimestamp = 1006;
    optional uint32 RxByteRate = 1007;
    optional uint32 TxByteRate = 1008;
    optional uint32 RxAvgByteRate = 1009;
    optional uint32 TxAvgByteRate = 1010;
    optional uint32 RxError = 1011;
    optional uint32 TxError = 1012;
```

```
    optional uint32 ReassocCount = 1013;
    optional uint32 TxRetryBytes = 1014;
    optional uint32 RxDropPkts = 1015;
    optional int32 vni = 1016;
    optional uint32 authStatus = 1017;
    optional string authMethod = 1018;
    optional string encryption = 1019;
    optional uint32 disconnectReason = 1020;
}

message APClientWlan {
    required string ssid = 1;
    optional string bssid = 2;
    optional int32 vlan = 3;
    optional int32 wsgWlanId = 4;
    optional int32 wlanId = 5;
    optional string wlangroup_id = 6;
    optional string wlantenant_id = 7;
    optional string wlangroup_name = 8;
    optional string wlantenant_name = 9;
}
message APClientRadio {
    required int32 radioId = 1;
    optional string mode = 2;
    optional string radioMode = 3;
    optional int32 channel = 4;
    optional uint32 channelWidth = 5;
    optional APClientWlan wlan = 6;
}
message APClientStats {
    required uint32 version = 1;
    optional string ap = 2;
    optional uint64 timestamp = 3;
    optional uint64 seqNumber = 4;
    optional string zone_id = 5;
    repeated APClientInfo clients = 6;
    optional string apgroup_id = 7;
    optional string cluster_id = 8;
    optional string domain_id = 9;
    optional string aptenant_id = 10;
    optional string map_id = 11;
    optional string aptenant_name = 12;
    optional string zone_name = 13;
    optional string apgroup_name = 14;
    optional string domain_name = 15;
    optional uint64 sampleTime = 16;
    optional uint32 aggregationInterval = 17;
    optional string deviceName = 18;
    optional string serialNumber = 19;
    optional string apRadioDeploy = 20;
}
```

Field Description

TABLE 13 AP Client information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
clientMac	string	snapshot	NULL	MAC address of the client
ipAddress	string	snapshot	NULL	IP address of the client
ipv6Address	string	snapshot	NULL	IPv6 address of the AP
wlanId	int	snapshot	NULL	WLAN interface ID
rssi	int	snapshot	AVG	Last recorded RSSI/SNR

TABLE 13 AP Client information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
receiveSignalStrength	int	snapshot	AVG	Last recorded signal strength received.
noiseFloor	int	snapshot	AVG	Last recorded noise floor
vlan	uint64	snapshot	NULL	Client VLAN ID
rxFrames	uint64	delta	SUM	Data frames received
rxBytes	uint64	delta	SUM	Data count received (in bytes)
txFrames	uint64	delta	SUM	Data frames transmitted
txBytes	uint64	delta	SUM	Data count transmitted (in bytes)
txMgmtFrames	uint64	snapshot	SUM	Accumulated number of transmitted packets
rxMgmtFrames	uint64	snapshot	SUM	Accumulated number of received packets
throughputEst	uint32	snapshot	AVG	Current throughput
txDropDataFrames	uint64	snapshot	SUM	Total Tx data frames dropped or dropped by MQ. In AP internal design, it has a messages queue (MQ) to queue all packets that AP plans to forward to clients. The AP transmit packets to clients according to the priority and scheduling. The MQ is Ruckus proprietary.
txDropMgmtFrames	uint64	snapshot	SUM	Total transaction management frames dropped (or dropped by MQ)
rxCRCErrFrames	uint32	snapshot	SUM	Number of Rx frames with CRC errors
txRetry	uint32	snapshot	SUM	Number of retried frames
osType	string	snapshot	NULL	Client OS type,such as Windows7/ Vista, Android.
radio	.APClientRadio	serialization	NULL	Radio information
tcWithQuota	.APClientInfo.TCWith Quota	serialization	NULL	Traffic class stats per client
cpeMac	string	snapshot	NULL	MAC address for the client connected behind CPE
stickyWeak	unit32	snapshot	NULL	Client is sticky/weak client
deviceType	int32	snapshot	NULL	Client device type, such as Laptop, Smartphones, Tablets etc
osVendorType	int32	snapshot	NULL	Client OSVendor type, such as Windows, Android, Apple etc
modelName	string	snapshot	NULL	Client's device model name, such as Windows 7, Mac OS X etc
hostname	string	snapshot	NULL	Client hostname
medianTxMCSRate	unit32	snapshot	NULL	Client median TX MCS rate
medianRxMCSRate	unit32	snapshot	NULL	Client median RX MCS rate
default_gateway	string	snapshot	NULL	Default gateway of the client
dhcp_server_ip	string	snapshot	NULL	DHCP server ip address of the client
subnet_mask	string	snapshot	NULL	Subnet mask of the client

TABLE 13 AP Client information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
dns_server_list	string	snapshot	NULL	DNS server list of the client
mlo_capability	string	snapshot	NULL	Client capabilities of Multi Link Operation like EMLSR or EMLMR
mlo_links	string	snapshot	NULL	Client link of Multi Link Operation
mld_addr	string	snapshot	NULL	Client MLD Mac address
active_link	bool	snapshot	NULL	MLO Client Active link
ConnectMode	string	snapshot	NULL	Client wireless connection mode.
Username	string	snapshot	NULL	Client authorization username.
SessionId	string	snapshot	NULL	Client accounting session ID.
MultipleSessionId	string	snapshot	NULL	Client accounting multiple session ID.
AuthMode	string	snapshot	NULL	Client authentication mode.
DiscTimestamp	uint64	snapshot	NULL	Client disconnection timestamp.
RxAvgByteRate	uint32	snapshot	AVG	Client average receive data rate
TxAvgByteRate	uint32	snapshot	AVG	Client average transmission data rate
RxError	uint32	snapshot	Snapshot	Client RX CRC error count.
TxRetryBytes	uint32	snapshot	Snapshot	Client current TX retry count.
vni	int32	snapshot	Snapshot	Client Network Segmentation (MDU) VNI
authStatus	int32	snapshot	Snapshot	Wlan auth status
authMethod	string	snapshot	Snapshot	Wlan authentication method
encryption	string	snapshot	Snapshot	Encryption method used by the AP
disconnectReason	int32	snapshot	Snapshot	Reason for disconnect from the controller.

TABLE 14 AP Client Info .TCWithQuota

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
tcName	string	snapshot	NULL	Name of the traffic class.
tcMaxQuota	string	snapshot	NULL	Amount of quota assigned(x) in bytes, received during access accept.
tcRemainingQuota	string	snapshot	NULL	Bytes remaining for the client after consuming n bytes (x-n).

TABLE 15 AP Client WLAN

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ssid	string	snapshot	NULL	SSID of the WLAN
bssid	string	snapshot	NULL	BSSID of the WLAN
vlan	uint64	snapshot	NULL	Client VLAN ID
wsgWlanId	int	snapshot	NULL	Unique WLAN ID assigned by the controller

TABLE 15 AP Client WLAN (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
wlanId	int	snapshot	NULL	WLAN interface ID
wlangroup_id	string	snapshot	NULL	WLAN Group identifier
wlantenant_id	string	snapshot	NULL	WLAN tenant identifier
wlangroup_name	string	snapshot	NULL	WALN Group name
wlantenant_name	string	snapshot	NULL	WLAN tenant name

TABLE 16 AP Client Radio

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
radioId	int32	snapshot	NULL	Radio ID number used by the AP
mode	string	snapshot	NULL	Radio mode used by the AP's wireless interface
80211RadioMode	string	snapshot	NULL	Radio mode used by the AP
channel	int32	snapshot	NULL	Current radio channel used by the AP
channelWidth	uint32	snapshot	NULL	Channel width used by the AP's radio
wlan	.APClientWlan	snapshot	NULL	WLAN information for this client

TABLE 17 AP Client Stats

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
ap	string	snapshot	NULL	MAC address of the AP
timeStamp	string	snapshot	NULL	Date and time this report was generated
seqNumber	uint64_t	snapshot	NULL	Counter for generating mesh statistics. The count increases by one whenever the AP generates mesh statistics. When the AP restarts, the counter also resets to zero(0).
zoneUUID	string	snapshot	NULL	Unique zone ID (for example, b381206b-2e5d-43dc-b249-e36ffae9855c) assigned by the controller
clients	.APClientInfo	Serialization	NULL	AP current client list
apgroup_id	string	snapshot	NULL	AP group UUID
cluster_id	string	snapshot	NULL	Cluster UUID
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	Map UUID
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name
apgroup_name	string	snapshot	NULL	AP group name

TABLE 17 AP Client Stats (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
domain_name	string	snapshot	NULL	Domain name
sampleTime	uint64	snapshot	NULL	Timestamp for this stats report generation
aggregationInterval	uint32	snapshot	NULL	Interval time for data aggregation
deviceName	string	snapshot	NULL	AP device name
serialNumber	string	snapshot	NULL	The serial number in AP borad data
`apRadioDeploy`	string	snapshot	NULL	Indicate AP all radio band deploy

ap_hccd_report.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";
import "nanopb.proto";

message ApHccdClientReportConnection {
    optional string client_mac = 1;
    optional uint64 timestamp = 2;
    optional int32 connection_status = 3;
    optional int32 failed_msg_id = 4;
    repeated int32 message_ids = 5;
    optional uint32 wlan_id = 6;
    optional uint32 radio_id = 7;
    optional string ssid = 8;
    optional string wlanType = 9;
    optional float snr = 10;
    optional int32 failure_type = 11;
    optional int32 vlan = 12;
    optional int32 reason_code = 13;
    optional string info = 14;
    optional uint64 TT = 15;
    optional int32 isRoaming = 16;
    optional string pktCapName = 17;
}

message ApHccdReportMessage {
    optional uint32 version = 1;
    optional string ap_mac = 2;
    optional uint64 timestamp = 3;
    optional uint64 seqNumber = 4;
    optional string zone_id = 5;
    optional string apgroup_id = 6;
    optional string cluster_id = 7;
    optional string domain_id = 8;
    optional string aptenant_id = 9;
    optional string map_id = 10;
    optional string deviceName = 11;
    optional string aptenant_name = 12;
    optional string zone_name = 13;
    optional string apgroup_name = 14;
    optional string domain_name = 15;
    optional uint64 sampleTime = 16;
    optional uint32 aggregationInterval = 17;
    optional string apIpAddress = 18;
    optional string apIpv6Address = 19;
    repeated ApHccdClientReportConnection apHccdClients = 20 [(nanopb).type = FT_POINTER];
    optional string apRadioDeploy = 21;
}
```

Field Description

TABLE 18 AP HCCD Client Report Connection information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
client_mac	string	snapshot	NULL	Client MAC address

TABLE 18 AP HCCD Client Report Connection information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
timestamp	uint64	snapshot	NULL	Timestamp for starting to record this client session
connection_status	int32	snapshot	NULL	Client session connection status (success or failure)
failed_msg_id	int32	snapshot	NULL	Message ID to indicate what failures step in whole connection
message_ids	int32	Serialization	NULL	A sequence of message ID are recorded for this client session
wlan_id	uint32	snapshot	NULL	WLAN Id that is connected by client
radio_id	uint32	snapshot	NULL	Radio Id that is connected by client
ssid	string	snapshot	NULL	SSID that is connected by client
failure_type	uint32	snapshot	NULL	Classify this failure client session belong to which failure case (auth, assoc, eap, radio, dhcp)
vlan	uint32	snapshot	NULL	Vlan ID is used by client
reason_code	uint32	snapshot	NULL	Reason code for deauth/disassoc frame
info	string	snapshot	NULL	Client session disconnection description
TTF	uint64	snapshot	NULL	Time spent for this failure connection
isRoaming	int32	snapshot	NULL	Flag to indicate this connection session is roaming or new join
pktCapName	string	snapshot	NULL	CCD packet capture file name

TABLE 19 AP HCCD Report information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
version	uint32	snapshot	NULL	AP HCCD version
ap_mac	string	snapshot	NULL	AP MAC address
timestamp	uint64	snapshot	NULL	Timestamp to generate this stats report
seqNumber	uint64	snapshot	NULL	Sequence number to identify this stats report
zone_id	string	snapshot	NULL	Zone UUID
apgroup_id	string	snapshot	NULL	AP gorup UUID
cluster_id	string	snapshot	NULL	Cluster UUID
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	MAP UUID
deviceName	string	snapshot	NULL	AP device name
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name
apgroup_name	string	snapshot	NULL	AP group name

TABLE 19 AP HCCD Report information (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
domain_name	string	snapshot	NULL	Domain name
sampleTime	uint64	snapshot	NULL	Timestamp for generating this report
aggregationInterval	uint32	snapshot	NULL	Interval time for aggregrating stats data
apIpAddress	string	snapshot	NULL	AP IPv4 address
apIpv6Address	string	snapshot	NULL	AP IPv6 address
apHccdClients	.ApHccdClientReport Connection	Serialization	NULL	Serialization data for failure client session record
apRadioDeploy	string	snapshot	NULL	Indicate AP all radio band deploy

ap_mesh.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APMeshDownlink {
    required string downMac = 1;
    optional uint32 type = 2;
    optional int32 rssi = 3;
    optional uint64 txBytes = 4;
    optional uint64 txFrames = 5;
    optional uint64 rxBytes = 6;
    optional uint64 rxFrames = 7;
    optional uint32 medianTxMCSRate = 8;
    optional uint32 medianRxMCSRate = 9;
}

message APMeshUplink {
    required string upMac = 1;
    optional uint32 type = 2;
    optional int32 rssi = 3;
    optional uint64 txBytes = 4;
    optional uint64 txFrames = 5;
    optional uint64 rxBytes = 6;
    optional uint64 rxFrames = 7;
    optional uint32 medianTxMCSRate = 8;
    optional uint32 medianRxMCSRate = 9;
}

message APMeshNeighbor {
    required string mac = 1;
    optional int32 rssi = 2;
}

message APMeshStats {
    required uint32 version = 1;
    required string ap = 2;
    optional uint64 timestamp = 3;
    optional uint64 seqNumber = 4;
    optional string zone_id = 5;
    optional int32  meshRole = 6;
    optional string meshSSID = 7;
    optional string meshPassphraseMD5 = 8;
    optional int32  hops = 9;
    repeated APMeshNeighbor neighbor = 10;
    repeated APMeshUplink uplink = 11;
    repeated APMeshDownlink downlink = 12;
    optional string apgroup_id = 13;
    optional string cluster_id = 14;
    optional string domain_id = 15;
    optional string aptenant_id = 16;
    optional string map_id = 17;
    optional string aptenant_name = 18;
    optional string zone_name = 19;
    optional string apgroup_name = 20;
    optional string domain_name = 21;
    optional uint64 sampleTime = 22;
    optional uint32 aggregationInterval = 23;
    optional bool isMeshEnable = 24;
    optional string serialNumber = 25;
}
```

Field Description

TABLE 20 AP Mesh Downlink information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
downMac	string	snapshot	NULL	MAC address of the mesh downlink AP
type	uint32	snapshot	NULL	Link status of the mesh downlink AP
rssi	int32	snapshot	AVG	RSSI of the mesh downlink AP
txBytes	uint64	snapshot	SUM	Total size of data and management packets transmitted since the last AP restart
txFrams	uint64	snapshot	SUM	Total number of data and management packets transmitted since the last AP restart
rxBytes	uint64	snapshot	SUM	Total size of data and management packets received since the last AP restart
rxFrames	uint64	snapshot	SUM	Total number of data and management packets received since the last AP restart
medianTxMCSRate	UINT32	delta	NULL	Downlink median TX MCS rate in this bin
medianRxMCSRate	UINT32	delta	NULL	Downlink median RX MCS rate in this bin

TABLE 21 AP Mesh Uplink Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
upMac	string	snapshot	NULL	MAC address of the mesh uplink AP
type	uint32	snapshot	NULL	Link status of the mesh uplink AP
rssi	int32	snapshot	AVG	RSSI of the mesh uplink AP
txBytes	uint64	snapshot	SUM	Total size of data and management packets transmitted since the last AP restart
txFrams	uint64	snapshot	SUM	Total number of data and management packets transmitted since the last AP restart
rxBytes	uint64	snapshot	SUM	Total size of data and management packets received since the last AP restart
rxFrames	uint64	snapshot	SUM	Total number of data and management packets received since the last AP restart
medianTxMCSRate	UINT32	delta	NULL	Uplink median TX MCS rate in this bin
medianRxMCSRate	UINT32	delta	NULL	Uplink median RX MCS rate in this bin

TABLE 22 AP Mesh Neighbor Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
mac	string	snapshot	NULL	MAC address of the neighboring AP
rssi	int32	snapshot	NULL	RSSI of the neighboring AP

TABLE 23 AP Mesh Statistics Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
version	uint32	snapshot	NULL	Mesh stats version
ap	string	snapshot	NULL	MAC address of theAP
timeStamp	string	snapshot	NULL	Date and time these statistics were generated
seqNumber	uint64	snapshot	NULL	Counter for generating mesh statistics.The count increases by one whenever the AP generates mesh statistics.When the AP restarts,the counter also resets to zero(0).
zoneUUID	string	snapshot	NULL	Unique zone ID (for example, b381206b-2e5d-43dc-b249-e36ffae9855c) assigned by the controller
meshRole	int	snapshot	NULL	Role of the AP on the mesh network (if mesh networking is enabled). Possible values include: • 0: MESH_DISABLED • 1: MESH_RAP • 2: MESH_MAP • 3: MESH_EAP • 4: MESH_DOWN • 5: MESH_UNDEFINED
meshSSID	int	snapshot	NULL	SSID of the mesh network
meshPassphraseMD5	int	snapshot	NULL	Passphrase required for the AP to join the mesh network
hops	int	snapshot	NULL	Depth or number of hops between this AP and its parent root AP
neighbor	.APMeshNeighbor	Serialization	NULL	Serialization neighbors data
uplink	.APMeshUplink	Serialization	NULL	Serialization mesh uplink data
downlink	.APMeshDownlink	Serialization	NULL	serialization mesh downlink data
apgroup_id	string	snapshot	NULL	AP group UUID
cluster_id	string	snapshot	NULL	Cluster UUID
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	MAP UUID
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name
apgroup_name	string	snapshot	NULL	AP group name

TABLE 23 AP Mesh Statistics Information (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
domain_name	string	snapshot	NULL	Domain name
sampleTime	uint64	snapshot	NULL	Timestamp for generating this stats report
aggregationInterval	uint32	snapshot	NULL	Interval time to aggregate data
isMeshEnable	boolean	snapshot	NULL	An indicate for mesh status
serialNumber	string	snapshot	NULL	The serial number in AP borad data

ap_peerlist.proto

```
/**
 * Copyright 2013 Ruckus Wireless, Inc. All rights reserved.
 *
 * RUCKUS WIRELESS, INC. CONFIDENTIAL -
 * This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is
 * fully protected under copyright and trade secret laws. You may not view,
 * use, disclose, copy, or distribute this file or any information contained
 * herein except pursuant to a valid license from Ruckus.
 */
option java_package = "com.ruckuswireless.scg.protobuf";

message APPeer {
    optional string ip_learned = 1;
    required string basemac = 2;
    optional uint64 rx = 3;
    optional uint64 tx = 4;
    optional uint64 timestamp = 5;
    optional string stat = 6;
    optional uint64 last_rx_from_now = 7;
    optional uint32 rssi0 = 8;
    optional uint32 rssi0x = 9;
    optional uint32 rssi1 = 10;
    optional uint32 rssi1x = 11;
    optional uint32 chan0 = 12;
    optional uint32 chan1 = 13;
    optional uint32 lastupdate = 14;
    optional string conn_lrv = 15;
    optional string conn_ip = 16;
    optional string conn_status = 17;
    optional uint64 conn_lastupdate = 18;
    optional string ipv6_learned = 19;
    optional uint32 rssi2 = 20;
    optional uint32 rssi2x = 21;
    optional uint32 chan2 = 22;
    optional uint32 distance = 23;

}

message APPeerReport {
    required uint32 version = 1;
    optional string apMac = 2;
    optional uint64 timestamp = 3;
    optional uint64    seqNumber = 4;
    optional string    zone_id = 5;
    repeated APPeer peerstat = 6;
    optional string apgroup_id = 7;
    optional string cluster_id = 8;
    optional string domain_id = 9;
    optional string aptenant_id = 10;
    optional string map_id = 11;
    optional string aptenant_name = 12;
    optional string zone_name = 13;
    optional string apgroup_name = 14;
    optional string domain_name = 15;
       optional string apRadioDeploy = 16;
       optional AttributeMap ideal_stats_required = 17;
       optional bool reboot = 18;
       optional string serialNumber = 19;
}
```

Field Description

TABLE 24 AP Report Bin Tunnel Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
type	int	snapshot	NULL	Tunnel type: 0: rks_gre 1: soft_gre
apIPAddress	string	snapshot	NULL	AP IP address

ap_report.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APReportPerDnsServer {
    required string ipAddress = 1;
    optional uint64 req_cnt = 2;
    optional uint64 resp_cnt = 3;
}

message APReportDnsStats {
    required int32 server_cnt = 1;
    optional .APReportPerDnsServer perDnsServer = 2;
}

message APReportBinWlan {
    optional string ap_mac = 1;
    optional uint64 time = 2;
    optional uint64 binStartTime = 3;
    optional string radioMode = 4;
    optional string mode = 5;
    optional int32 channel = 6;
    optional string ssid = 7;
    optional string bssid = 8;
    optional int32 wsgWlanId = 9;
    optional int32 wlanId = 10;
    optional uint32 maxNumClients = 11;
    optional uint32 minNumClients = 12;
    optional int32 avgNumClients = 13;
    optional uint64 rxBytes_r = 14;
    optional uint64 txBytes_r = 15;
    optional uint64 rxFrames_r = 16;
    optional uint64 txFrames_r = 17;
    optional uint64 txFail_r = 18;
    optional uint64 rxRateKbps = 19;
    optional uint64 txRateKbps = 20;
    optional uint32 newAssoc = 21;
    optional uint32 failedAssoc = 22;
    optional uint32 rxFail_r = 23;
    optional uint64 peakRx_r = 24;
    optional uint64 peakTx_r = 25;
    optional uint64 rxDataFrames_r = 26;
    optional uint64 rxDataBytes_r = 27;
    optional uint64 rxMgmtFrames_r = 28;
    optional uint64 rxMgmtBytes_r = 29;
    optional uint64 txDataFrames_r = 30;
    optional uint64 txDataBytes_r = 31;
    optional uint64 txMgmtFrames_r = 32;
    optional uint64 txMgmtBytes_r = 33;
    optional uint64 rxBcastFrames_r = 34;
    optional uint64 rxMcastFrames_r = 35;
    optional uint64 rxUcastFrames_r = 36;
    optional uint64 txBcastFrames_r = 37;
    optional uint64 txMcastFrames_r = 38;
    optional uint64 txUcastFrames_r = 39;
    optional uint64 txDropDataFrames_r = 40;
    optional string wlangroup_id = 42;
    optional string wlantenant_id = 43;
    optional string wlangroup_name = 44;
    optional string wlantenant_name = 45;
    optional string wlanName = 46;
    optional uint32 authFailureCount = 47;
    optional uint32 authSuccessCount = 48;
```

```
    optional uint32 assocFailureCount = 49;
    optional uint32 assocSuccessCount = 50;
    optional uint32 eapFailureCount = 51;
    optional uint32 eapSuccessCount = 52;
    optional uint32 radiusFailureCount = 53;
    optional uint32 radiusSuccessCount = 54;
    optional uint32 dhcpFailureCount = 55;
    optional uint32 dhcpSuccessCount = 56;
    optional uint64 txDropMgmtFrames_r = 57;
    optional uint32 staSmartRoamDisconCnt = 58;
    optional uint32 staIdleDisconCnt = 59;
    optional uint32 staLeaveDisconCnt = 60;
    optional uint32 staInvalidDisconCnt = 61;
    optional uint32 staRadioFailDisconCnt = 62;
    optional uint32 staAPKickDisconCnt = 63;
    optional uint64 rxDataBytesSplitTunnel = 64;
    optional uint64 txDataBytesSplitTunnel = 65;
    optional uint64 rxDataFramesSplitTunnel = 66;
    optional uint64 txDataFramesSplitTunnel = 67;
    optional .APReportDnsStats dnsStats = 68;
    optional uint32 roamingFailureCount = 69;
    optional uint32 L3authFailureCount = 70;
    optional uint32 L3authSuccessCount = 71;
    optional uint32 roamingSuccessCount = 72;
    optional uint32 arpPktCount = 73;
    optional AttributeMapWlan ideal_stats_wlan_required = 74;
    optional uint32 probe_dropped = 75;
    optional uint32 auth_dropped = 76;
    optional uint32 btm_requests = 77;
    optional uint32 sta_kickouts = 78;
    optional uint32 rx_num_probe_filt = 79;
    optional uint32 rx_num_auth_filt = 80;
}

message APReportBinRadio {
    optional uint32 radioId = 1;
    optional uint32 airtime = 2;
    optional uint32 airtimeB = 3;
    optional uint32 airtimeRx = 4;
    optional uint32 airtimeTx = 5;
    optional uint32 phyError = 6;
    optional uint64 rxBytes_r = 7;
    optional uint64 txBytes_r = 8;
    optional uint64 rxFrames_r = 9;
    optional uint64 txFrames_r = 10;
    optional int32  noiseFloor = 11;
    optional uint64 retry = 12;
    repeated APReportBinWlan binWlan = 13;
    optional uint32 latency = 14;
    optional uint32 capacity = 15;
    optional float connectionFailure = 16;
    optional uint32 connectionAuthFailureCount = 17;
    optional uint32 connectionAssocFailureCount = 18;
    optional uint32 connectionTotalCount = 19;
    optional uint32 connectionAuthSuccessCount = 20;
    optional uint32 connectionAssocSuccessCount = 21;
    optional uint32 connectionEAPFailureCount = 22;
    optional uint32 connectionEAPSuccessCount = 23;
    optional uint32 connectionRadiusFailureCount = 24;
    optional uint32 connectionRadiusSuccessCount = 25;
    optional uint32 connectionDHCPFailureCount = 26;
    optional uint32 connectionDHCPSuccessCount = 27;
    optional uint32 connectionTotalSuccess = 28;
    optional uint32 connectionTotalFailure = 29;
    optional uint64 txRatebps = 30;
    optional uint64 histogramClientTTC = 31;
    optional uint32 medianTxRadioMCSRate = 32;
    optional uint32 medianRxRadioMCSRate = 33;
    optional uint32 connectionL3AuthFailureCount = 34;
    optional uint32 connectionL3AuthSuccessCount = 35;
    optional uint32 TxPER = 36;
}
```

```
message APReportBinTunnel {
    required string gw = 1;
    optional int32 index = 2;
    optional int32 isActive = 3;
    optional uint64 cICMP = 4;
    optional uint64 cNonICMP = 5;
    optional uint64 cDisconnect = 6;
    optional uint64 rxBytes = 7;
    optional uint64 rxPkts = 8;
    optional uint64 rxDropPkts = 9;
    optional uint64 rxErrPkts = 10;
    optional uint64 txBytes = 11;
    optional uint64 txPkts = 12;
    optional uint64 txDropPkts = 13;
    optional uint64 txErrPkts = 14;
    optional uint64 txFragPkts = 15;
    optional int32 type = 16;
    optional string apIpAddress = 17;
}
message APReportBinIPSec {
      enum IPsecTunnelType {
        RUCKUSGRE = 0;
        SOFTGRE = 1;
    }
    required uint64 ipsecSessionTime = 1;
    optional uint64 ipsecTxPkts = 2;
    optional uint64 ipsecRxPkts = 3;
    optional uint64 ipsecTxBytes = 4;
    optional uint64 ipsecRxBytes = 5;
    optional uint64 ipsecTxDropPkts = 6;
    optional uint64 ipsecRxDropPkts    = 7;
    optional uint64 ipsecTxIdleTime    = 8;
    optional uint64 ipsecRxIdleTime    = 9;
    optional string apIpAddress = 10;
    optional string gw = 11;
    optional IPsecTunnelType ipsecTunnelType = 12;
}
message ttcData {
    optional string sessionId = 1;
    optional string multiSessionId = 2;
    optional bool isRoaming = 3;
    optional uint64 clientAuthTTC = 4;
    optional uint64 clientAssocTTC = 5;
    optional uint64 clientEapTTC = 6;
    optional uint64 clientRadiusTTC = 7;
    optional uint64 clientDhcpTTC = 8;
    optional bool isDHCPrenewSess = 9;

}

message sessDeauthData {
    optional string sessionId = 1;
    optional string multiSessionId = 2;
    optional uint64 disconnectTime = 3;
}

message APReportBinClient {
    required string ap = 1;
    optional uint64 time = 2;
    optional uint64 binStartTime = 3;
    optional string radioMode = 4;
    optional string ap80211RadioMode = 5;
    optional string auth = 6;
    optional string encryption = 7;
    optional string clientMac = 8;
    optional string bssid = 9;
    optional string ssid = 10;
    optional string username = 11;
    optional string clientIP = 12;
    optional uint64 clientVlan = 13;
    optional string osType = 14;
```

```
    optional string hostname = 15;
    optional int32 channel = 16;
    optional uint32 channelWidth = 17;
    optional int32 rssi = 18;
    optional int32 maxRssi = 19;
    optional int32 minRssi = 20;
    optional int32 firstRssi = 21;
    optional int32 receiveSignalStrength = 22;
    optional int32 firstReceiveSignalStrength = 23;
    optional int32 maxReceiveSignalStrength = 24;
    optional int32 minReceiveSignalStrength = 25;
    optional int32 noiseFloor = 26;
    optional string location = 27;
    optional uint64 rxBytes_r = 28;
    optional uint64 txBytes_r = 29;
    optional uint64 rxFrames_r = 30;
    optional uint64 txFrames_r = 31;
    optional uint32 throughputEst = 32;
    optional uint64 firstSampleTime = 33;
    optional uint64 txDropDataFrames_r = 35;
    optional uint64 rxCRCErrFrames_r = 36;
    optional string sessionId = 37;
    optional string multiSessionId = 38;
    optional uint64 firstConnection    = 39;
    optional uint64 firstAuth = 40;
    optional uint64 ipAssignTime = 41;
    optional uint64 disconnectTime = 42;
    optional uint64 sessionTime = 43;
    optional uint32 radioId = 44;
    optional int32 wsgWlanId = 45;
    optional string wlangroup_id = 46;
    optional string wlangroup_name = 47;
    optional uint32 disconnectReason = 48;
    optional string wlanName = 49;
    optional string wlantenant_id = 50;
    optional string wlantenant_name = 51;
    optional uint64 rxBytes = 52;
    optional uint64 txBytes = 53;
    optional uint64 rxRatebps = 54;
    optional uint64 txRatebps = 55;
    optional int32 bandCap = 56;
    optional int32 vHTCap = 57;
    optional int32 streamCap = 58;
    optional int32 BTMCap = 59;
    optional uint32 medianTxMCSRate = 60;
    optional uint32 medianRxMCSRate = 61;
    optional uint64 clientAuthTTC = 62;
    optional uint64 clientAssocTTC = 63;
    optional uint64 clientEapTTC = 64;
    optional uint64 clientRadiusTTC = 65;
    optional uint64 clientDhcpTTC = 66;
    optional uint32 roamingFailureCount = 67;
    optional uint32 roamingSuccessCount = 68;
    optional int32 deviceType = 69;
    optional int32 osVendorType = 70;
    optional string modelName = 71;
    optional uint32 TxPER = 72;
    optional string authMethod = 73;
    repeated ttcData clientTTC = 74;
    repeated sessDeauthData sessDeauthTime = 75;
    optional int32 WiFi6Cap = 76;
}

message HccdConnMessage {
    optional uint64 timestamp = 1;
    optional int32 message_id = 2;
    optional int32 source = 3;
    optional int32 destination = 4;
    optional int32 status_code = 5;
}

message HccdClientConnection {
```

```
    optional string client_mac = 1;
    optional uint64 timestamp = 2;
    optional int32 connection_status = 3;
    /* remove client_sm_map */
    optional int32 failed_msg_id = 5;
    repeated HccdConnMessage hccdConnMessages = 6;
    optional uint32 wlan_id = 7;
    optional uint32 radio_id = 8;
        optional string ssid = 9;
        optional string wlanType = 10;
        optional float snr = 11;
        optional int32 failure_type = 12;
    optional int32 vlan = 13;
        optional int32 reason_code = 14;
        optional string info = 15;
}

message APReportBin {
    required int32 bin = 1;
    optional int32 uptime_r = 2;
}

message APFirewallProfileStats {
    optional string firewallProfileId = 1;
    optional string firewallProfileName = 2;
    optional uint64 firewallProfileHitCount = 3;
    optional string l2AclPolicyId = 4;
    optional uint64 l2AclPolicyHitCount = 5;
    optional string l3AclPolicyId = 6;
    optional uint64 l3AclPolicyHitCount = 7;
    optional string devicePolicyId = 8;
    optional uint64 devicePolicyHitCount = 9;
    optional string avcPolicyId = 10;
    optional uint64 avcPolicyHitCount = 11;
    optional string urlFilteringId = 12;
    optional uint64 urlFilteringHitCount = 13;
}

message APReportStats {
    required uint32 version = 1;
    optional string ap = 2;
    optional uint64 timestamp = 3;
    optional uint64 seqNumber = 4;
    optional string zone_id = 5;
    repeated APReportBin binCount = 6;
    repeated APReportBinClient binClient = 7;
    repeated APReportBinIPSec binIPSec = 8;
    repeated APReportBinTunnel binTunnel = 9;
    repeated APReportBinRadio binRadio = 10;
    optional string deviceName = 11;
    optional string apgroup_id = 12;
    optional string cluster_id = 13;
    optional string domain_id = 14;
    optional string aptenant_id = 15;
    optional string map_id = 16;
    optional string aptenant_name = 17;
    optional string zone_name = 18;
    optional string apgroup_name = 19;
    optional string domain_name = 20;
    optional uint64 sampleTime = 21;
    optional uint32 aggregationInterval = 22;
    optional string apIpAddress = 23;
    optional string apIpv6Address = 24;
    repeated HccdClientConnection hccdClientConnections = 25;
                optional bool isMonitoringEnabled = 29;
    repeated APFirewallProfileStats firewallProfileStats = 30;
    optional string apRadioDeploy = 31;
    optional AttributeMap ideal_stats_required = 32;
    optional uint64 rflowMaxCount = 33;
}
```

Field Description

TABLE 25 AP Report Per Dns Server Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
ipAddress	string	snapshot	NULL	DNS IP address
req_cnt	uint64	snapshot	NULL	DNS Request Counter
resp_cnt	uint64	snapshot	NULL	DNS Response Counter

TABLE 26 AP Report Dns Stats Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
ipsecSessionTime	uint32	snapshot	NULL	DNS Server Count per wlan
ipsecTxPkts	.APReportPerDnsServer	snapshot	MAXNULL	Per DNS Server Statistics

TABLE 27 AP Report Bin WLAN Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
ap_mac	string	snapshot	NULL	MAC address of the AP
time	uint64_t	snapshot	NULL	First sample time in this bin. The AP samples stats every 90 seconds internally, when AP boots up. So if AP boots up at 10:08:00, then AP gets stats at 10:09:30, 10:11:00, 10:12:30, 10:14:00, 10:15:30. So the time is 10:09:30 in Bin (10:00~10:15), and the time is 10:15:30 in Bin (10:15~10:30)
binStartTime	uint64	snapshot	NULL	Bin start time. The start timestamp of each Bin. For example it is 00:00:00 in Bin1 (00:00~00:15), and it is 00:15:00 in Bin2(00:15~00:30)...
80211RadioMode	string	snapshot	NULL	Radio mode used by the AP
mode	string	snapshot	NULL	Radio mode used by the AP
channel	int	snapshot	NULL	Radio channel used by the AP
ssid	string	snapshot	NULL	SSID of the WLAN
bssid	string	snapshot	NULL	BSSID of the WLAN
wsgWlanId	int	snapshot	NULL	WLAN ID assigned by the controller
wlanId	int	snapshot	NULL	WLAN interface ID
maxNumClients	uint32	snapshot	MAX	Highest number of clients during this sampling period
minNumClients	uint32	snapshot	MIN	Lowest number of clients during this sampling period
avgNumClients	int	snapshot	AVG	Average number of clients during this sampling period

TABLE 27 AP Report Bin WLAN Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
rxBytes_r	uint64	delta	SUM	Total size of data and management packets received
rxFrames_r	uint64	delta	SUM	Total size of data and management frames received
txFrames_r	uint64	delta	SUM	Total size of data and management frames transmitted
txFail_r	uint64	snapshot	SUM	Total number of Tx errors
rxRateKbps	uint64	delta	SUM	Rx rate
txRateKbps	uint64	delta	SUM	Tx rate
newAssoc	uint32	Delta	AVG	Number of new client associations during this sampling period
failedAssoc	uint32	Delta	SUM	Failed associate number in bin period
rxFail_r	uint32	snapshot	SUM	No space in Linux buffers
peakRx_r	uint64	Delta	SUM	Total delta bytes in one din period for RX data and mgmt. frame
peakTx_r	uint64	Delta	SUM	Total delta bytes in one din period for TX data and mgmt.frame
rxDataFrames_r	uint64	snapshot	SUM	Accumulate Rx packet number
rxDataBytes_r	uint64	snapshot	SUM	Accumulate Rx data bytes
rxMgmtFrames_r	uint64	snapshot	SUM	Accumulate Rx packet number
rxMgmtBytes_r	uint64	snapshot	SUM	Accumulate Rx mgmt. bytes
txDataFrames_r	uint64	snapshot	SUM	Accumulate Tx packet number
txDataBytes_r	uint64	snapshot	SUM	Accumulate Tx data bytes
txMgmtFrames_r	uint64	snapshot	SUM	Accumulate Tx packet number
txMgmtBytes_r	uint64	snapshot	SUM	Accumulate Tx mgmt. bytes
rxBcastFrames_r	uint64	snapshot	SUM	Broadcast packets received
rxMcastFrames_r	uint64	snapshot	SUM	Multicast packets received
rxUcastFrames_r	uint64	snapshot	SUM	Received data packets that does not include bcast and multicast
txBcastFrames_r	uint64	snapshot	SUM	Broadcast packets transmitted
txMcastFrames_r	uint64	snapshot	SUM	Multicast packets transmitted
txUcastFrames_r	uint64	snapshot	SUM	Transmit data packets that does not include bcast and multicast
txDropDataFrames_r	uint64	snapshot	SUM	Tx data frames that are dropped or dropped by MQ
txDropMgmtFrames_r	uint64	snapshot	SUM	Total Tx data frames dropped (or dropped by MQ. In AP internal design, it has a messages queue (MQ)to queue all packets that AP plans to forward to clients. Then AP transmit packets to clients as per the priority scheduling. The MQ is Ruckus proprietary internal design.
wlangroup_id	string	snapshot	NULL	WLAN Group identifier.
wlantenant_id	string	snapshot	NULL	WLAN tenant identifier.

TABLE 27 AP Report Bin WLAN Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
wlangroup_name	string	snapshot	NULL	WLAN Group name.
wlantenant_name	string	snapshot	NULL	WLAN tenant name.
wlaneName	string	snapshot	NULL	WLAN name.
authFailureCount	uint32	Delta	SUM	802.11 authentication failure count on WLAN (delta value)
authSuccessCount	uint32	Delta	SUM	802.11 authentication success count on WLAN (delta value)
assocFailureCount	uint32		SUM	802.11 association failure count on WLAN (delta value)
assocSuccessCount	uint32	Delta	SUM	802.11 association success count on WLAN (delta value)
eapFailureCount	uint32	Delta	SUM	EAP authentication failure count on WLAN (delta value)
eapSuccessCount	uint32	Delta	SUM	EAP authentication success count on WLAN (delta value)
radiusFailureCount	uint32	Delta	SUM	Radio failure count on WLAN (delta value)
radiusSuccessCount	uint32	Delta	SUM	Radio success count on WLAN (delta value)
dhcpFailureCount	uint32	Delta	SUM	DHCP failure count on WLAN (delta value)
dhcpSuccessCount	uint32	Delta	SUM	DHCP success count on WLAN (delta value)
staSmartRoamDisconCnt	uint32	Delta	SUM	Client disconnect count with smart-roaming reason
staIdleDisconCnt	uint32	Delta	SUM	Client disconnect count with idle reason
staLeaveDisconCnt	uint32	Delta	SUM	Client disconnect count with client active leave bss reason
staInvalidDisconCnt	uint32	Delta	SUM	Client disconnect count with client's invalid frame contents reason
staRadioFailDisconCnt	uint32	Delta	SUM	Client disconnect count with AP radio related reason
staAPKickDisconCnt	uint32	Delta	SUM	Client disconnect count with AP active kick out reason
rxDataBytesSplitTunnel	uint64	snapshot	SUM	Split-tunnel total data Bytes received
txDataBytesSplitTunnel	uint64	snapshot	SUM	Split-tunnel total data Bytes transmitted
rxDataFramesSplitTunnel	uint64	snapshot	SUM	Split-tunnel total data-frames received
txDataFramesSplitTunnel	uint64	snapshot	SUM	Split-tunnel total data-frames transmitted
dnsStats	.APReportDnsStats	snapshot	NULL	DNS statistics
roamingFailureCount	uint32	Delta	SUM	roaming faliure count on wlan(delta value)

TABLE 27 AP Report Bin WLAN Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
L3authFailureCount	uint32	Delta	SUM	L3 auth faliure count on wlan(delta value)
L3authSuccessCount	uint32	Delta	SUM	L3 auth success count on wlan(delta value)
roamingSuccessCount	uint32	Delta	SUM	Roaming Success count on wlan(delta value)
arpPktCount	uint32	Delta	SUM	Roaming Success count on wlan(delta value)
ideal_stats_wlan_required	AttributeMapWlan	snapshot	NULL	Ideal stats wlan required for ACX APs
probe_dropped	uint32	Delta	SUM	Probes dropped due to RSSI/SNR/PPS/ATD/TCM Threshold features
auth_dropped	uint32	Delta	SUM	duth reqs dropped due to RSSI/SNR/PPS/ATD/TCM Threshold features
btm_requests	uint32	Delta	SUM	11v BTM requests sent by AP for 11v Steering features (CLB, BB, SROAM)
sta_kickouts	uint32	Delta	SUM	STA Kicked due to Smart Roam Algorithm
rx_num_probe_filt	uint32	Delta	SUM	Probes dropped due to Smart Roam Algorithm
rx_num_auth_filt	uint32	Delta	SUM	Auth requests dropped due to Smart Roam Algorithm

TABLE 28 AP Report Bin Radio Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
airtime	uint32	snapshot	AVG	Exponential average of total channel utilization
airtimeB	uint32	snapshot	AVG	Exponential average percentage of channel busy time
airtimeRx	uint32	snapshot	AVG	Exponential average of channel availability for receiving
airtimeTx	uint32	snapshot	AVG	Exponential average of channel availability for transmitting
phyError	uint32	snapshot	SUM	Accumulated number of Rx physical errors
rxBytes_r	uint64	Delta	SUM	Total data bytes received on radio
txBytes_r	uint64	Delta	SUM	Total data bytes transmitted on radio
rxFrames_r	uint64	Delta	SUM	Total number of data frames received
txFrames_r	uint64	Delta	SUM	Total number of data frames transmitted
noiseFloor	int	snapshot	AVG	Last recorded noise floor
radioId	uint32	snapshot	snapshot	Radio ID (0: 2.4G 1: 5G)

TABLE 28 AP Report Bin Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
binWlan	.APReportBinWlan	Serialization	snapshot	Serialization data for all of wlan information
txRatebps	uint64	Delta	AVG	Radio average transmission rate
retry	uint64	SUM	NULL	Number of transmission retries
latency	uint32	snapshot	SUM	The time taken by a packet from Ethernet ingress to Radio egress or Tx complete
capacity	uint32	snapshot	SUM	The saturated throughput estimate of a link
connectionFailure	float	snapshot	SUM	The rate of client connection failure
connectionAuthFailureCount	uint32	Delta	SUM	802.11 authentication failure count on radio (delta value)
connectionAssocFailureCount	uint32	Delta	SUM	802.11 association failure count on radio (delta value)
connectionTotalCount	uint32	Delta	SUM	total client connection on radio, include success and failure counts(delta value)
connectionAuthSuccessCount	uint32	Delta	SUM	802.11 authentication success count on radio (delta value)
connectionAssocSuccessCount	uint32	Delta	SUM	802.11 association success count on radio (delta value)
connectionEAPFailureCount	uint32	Delta	SUM	EAP authentication failure count on radio (delta value)
connectionEAPSuccessCount	uint32	Delta	SUM	EAP authentication success count on radio (delta value)
connectionRadiusFailureCount	uint32	Delta	SUM	Radio failure count on radio (delta value)
connectionRadiusSuccessCount	uint32	Delta	SUM	Radio success count on radio (delta value)
connectionDHCPFailureCount	uint32	Delta	SUM	DHCP failure count on radio (delta value)
connectionDHCPSuccessCount	uint32	Delta	SUM	DHCP success count on radio (delta value)
connectionTotalSuccess	uint32	Delta	SUM	Total count for success connection (delta value)
connectionTotalFailure	uint32	Delta	SUM	Total count for failure connection (delta value)
txRatebps	uint64	Delta	AVG	Radio average transmission rate
histogramClientTTC	uint64	Delta	NULL	Client TTC histogram data count on radio
medianTxRadioMCSRate	uint32	Delta	NULL	Radio median TX MCS rate in this bin
medianRxRadioMCSRate	uint32	Delta	NULL	Radio median RX MCS rate in this bin
connectionL3AuthFailureCount	uint32	Delta	SUM	L3 auth failure count on radio (delta value)
connectionL3AuthSuccessCount	uint32	Delta	SUM	L3 auth success count on radio (delta value)

TABLE 28 AP Report Bin Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
TxPER	uint32	Delta	NULL	Radio Tx Packet Error Rate in this bin

TABLE 29 AP Report Bin Tunnel Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
gw	string	snapshot	NULL	Tunnel gateway address list
index	int	snapshot	NULL	Current tunnel index
isActive	int	snapshot	NULL	Current tunnel state (active or inactive)
cICMP	uint64	snapshot	MAX	Number of keepalive packets sent
cNonICMP	uint64	snapshot	MAX	Number of keepalive packets lost
cDisconnect	uint64	snapshot	MAX	Number of SoftGRE sessions terminated
rxBytes	uint64	snapshot	SUM	Total bytes received
rxPkts	uint64	snapshot	SUM	Total packets received
rxDropPkts	uint64	snapshot	SUM	No space in Linux buffers (frame equal to packet). The AP received a frame completely, and AP has to allocate a memory for this frame. Then pass to next handler who is interested in this frame. The AP drops the frame if it has not enough memory. The counter is plus one if this case happen.
rxErrPkts	uint64	snapshot	SUM	Number of bad packets received
txBytes	uint64	snapshot	SUM	Total bytes transmitted
txPkts	uint64	snapshot	SUM	Total packets transmitted
txDropPkts	uint64	snapshot	SUM	No space available in Linux. AP has to allocate a memory to store the packet that AP plans to transmit to client. The AP may drop it if AP has not enough memory.Then the counter is plus one.
txErrPkts	uint64	snapshot	SUM	Packet transmit problems. The AP plans to transmit packet to client. But somehow AP can't transmit to client successfully, and AP drops the packets at the end. Then the counter is plus one.
txFragPkts	uint64	snapshot	SUM	Total fragmented Tx packets
type	int	snapshot	NULL	Tunnel type: 0: rks_gre 1: soft_gre
apIPAddress	string	snapshot	NULL	AP IP address

TABLE 30 AP Report Bin IPSec Information

Name	Value	Description
RUCKUSGRE	0	Tunnel type possible value
SOFTGRE	1	Tunnel type possible value

TABLE 31 AP Report Bin IPSec Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
ipsecSessionTime	uint64	snapshot	MAX	Session time
ipsecTxPkts	uint64	snapshot	MAX	Total packets transmitted
ipsecRxPkts	uint64	snapshot	MAX	Total packets received
ipsecTxBytes	uint64	snapshot	MAX	Total bytes transmitted
ipsecRxBytes	uint64	snapshot	MAX	Total bytes received
ipsecTxDropPkts	uint64	snapshot	MAX	Total Tx packets dropped
ipsecRxDropPkts	uint64	snapshot	MAX	Total Rx packets dropped
ipsecTxIdleTime	uint64	snapshot	MAX	Tx idle time
ipsecRxIdleTime	uint64	snapshot	MAX	Rx idle time
apIPAddress	string	snapshot	NULL	AP IP address
gw	string	snapshot	NULL	Tunnel gateway address list
ipsecTunnelType	.APReportBinIPSec.IP secTunnelType	snapshot	NULL	Tunnel type: (0: rks_gre 1: soft_gre)

TABLE 32 ttcData Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
sessionId	string	snapshot	NULL	Session ID string
multiSessionId	string	snapshot	NULL	Multi-session ID string
isRoaming	bool	snapshot	NULL	Roaming session or not
clientAuthTTC	unti64	snapshot	NULL	Client auth TTC latest value in this bin.
clientAssocTTC	unti64	snapshot	NULL	Client assoc TTC latest value in this bin.
clientEapTTC	unti64	snapshot	NULL	Client EAP TTC latest value in this bin.
clientRadiusTTC	unti64	snapshot	NULL	Client Radius TTC latest value in this bin.
clientDhcpTTC	unti64	snapshot	NULL	Client DHCP TTC latest value in this bin.
isDHCPrenewSess	BOOL	snapshot	NULL	Is dhcp renew session

TABLE 33 Sess Deauth Data Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
sessionId	string	snapshot	NULL	Session ID string.
multiSessionId	string	snapshot	NULL	Multi-sesson ID string.

TABLE 33 Sess Deauth Data Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
disconnectTime	unti64	snapshot	NULL	Date and time client was disconnected.

TABLE 34 AP Report Bin Client Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ap	string	snapshot	NULL	MAC address of the AP
time	uint64	snapshot	NULL	Timestamp for data sampling time
binStartTime	uint64	snapshot	NULL	Bin start time The start timestamp of each Bin. For example it is 00:00:00 in Bin1 (00:00~00:15), and it is 00:15:00in Bin2 (00:15~00:30)...
radioMode	string	snapshot	NULL	Radio mode that is used by this client connection
ap80211RadioMode	string	snapshot	NULL	Radio mode used by the AP. Possible values are "b", "b/g", "b/g/n", "g", "g/n", "a", "a/n", "a/n/ac", "n", "n/ ac" , "ac"
auth	string	snapshot	NULL	Authorization mode used by the AP xxx what are the possible values?
encryption	string	snapshot	NULL	Encryption method used by the AP
clientMac	string	snapshot	NULL	MAC address of the client
bssid	string	snapshot	NULL	BSSID
ssid	string	snapshot	NULL	SSID
username	string	snapshot	NULL	User name
clientIP	string	snapshot	NULL	IP address assigned to the client
clientVlan	uint64	snapshot	NULL	VLAN ID used by the client
osType	string	snapshot	NULL	Operating system used by the client
hostname	string	snapshot	NULL	Host name of the client
channel	int	snapshot	NULL	Current radio channel
channelWidth	uint32	snapshot	NULL	Channel width used by the WLAN
rssi	int	snapshot	AVG	Last recorded RSSI/SNR
maxRssi	int	snapshot	MAX	Highest RSSI ever recorded
minRssi	int	snapshot	MIN	Lowest RSSI ever recorded
firstRssi	int	snapshot	AVG	Initial RSSI recorded
receiveSignalStrength	int	snapshot	AVG	Last received signal strength
firstReceiveSignalStrength	int	snapshot	MAX	Initial received signal strength
maxReceiveSignalStrength	int	snapshot	MIN	Highest received signal strength
minReceiveSignalStrength	int	snapshot	AVG	Lowest received signal strength
noiseFloor	int	snapshot	AVG	Last recorded noise floor
location	int	snapshot	NULL	Location of the AP
rxBytes_r	uint64	delta	SUM	Total bytes received

TABLE 34 AP Report Bin Client Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
txBytes_r	uint64	delta	SUM	Total bytes transmitted
rxFrames_r	uint64	delta	SUM	Data frames received
txFrames_r	uint64	delta	SUM	Data frames transmitted
throughputEst	uint64	delta	SUM	Average of non-zero throughput estimate avg_throughput_estimate = sum__throughput_estimate/ count_non_zero_throughput_estimat e For example AP has client's throughput estimation of 10Mbps, 9Mbps, 9Mbps, 10Mbps, 0, 0, 0, 0, 0, 0 . Then AP come out (10+9+9+10)/4 = 9.5Mbps for throughput Est.
firstSampleTime	uint64	snapshot	NULL	First sample time in this bin. The AP samples stats every 90 seconds internally, when AP boots up. So if AP boots up at 10:08:00. Then AP gets stats at 10:09:30, 10:11:00, 10:12:30, 10:14:00, 10:15:30. So the time is 10:09:30in Bin(10:00~10:15), and the time is10:15:30 in Bin(10:15~10:30)
txDropMgmtFrames_r	uint64	snapshot	SUM	Total Tx data frames dropped (or dropped by MQ. In AP internal design,it has a messages queue (MQ) to queue all packets that AP plans to forward to clients. Then AP transmit packets to clients as per the priority/ scheduling/... The MQ is Ruckus proprietary internal design.
txDropDataFrames_r	uint64	snapshot	SUM	Total Tx management frames dropped (or dropped by MQ. In AP internal design, it has a messages queue (MQ)to queue all packets that AP plans to forward to clients. Then AP transmit packets to clients as per the priority/scheduling/... The MQ is Ruckus proprietary internal design.
rxCRCErrFrames_r	uint64	snapshot	SUM	CRC error for Rx data frame. Each 802.11 frame has 4 bytes FCS at tail. The AP recalculate the value when it received a frame completely. Then compare to the FCS in the frame. AP drops the frame if they are different. Then counter plus 1.
sessionId	sting	snapshot	NULL	Session ID string
multiSessionId	sting	snapshot	NULL	Multi-session ID string
firstConnection	uint64	snapshot	NULL	Date and time of initial connection
firstAuth	uint64	snapshot	NULL	Date and time of initial authorization
ipAssignTime	uint64	snapshot	NULL	Date and time client IP address was assigned
disconnectTime	uint64	snapshot	NULL	Date and time client was disconnected

TABLE 34 AP Report Bin Client Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
sessionTime	uint64	snapshot	NULL	Duration of client session
radioId	uint32	snapshot	NULL	Radio interface identifier (0: 2.4G, 1 5G)
wsgWlanId	int	snapshot	NULL	WLAN ID assigned by the controller
wlangroup_id	string	snapshot	NULL	WLAN Group identifier
wlangroup_name	string	snapshot	NULL	WLAN Group name
disconnectReason	uint64	snapshot	NULL	Reason for disconnect from the controller.
wlanName	string	snapshot	NULL	WLAN name
wlantenant_id	string	snapshot	NULL	WLAN tenant identifier
wlantenant_name	string	snapshot	NULL	WLAN tenant name
rxBytes	uint64	snapshot	SUM	Total data bytes received for this client
txBytes	uint64	snapshot	SUM	Total data bytes transmitted by this client
rxRatebps	uint64	snapshot	AVG	Client receiving data rate in bin period
txRatebps	uint64	snapshot	AVG	Client transmitted data rate in bin period
bandCap	int32	snapshot	NULL	Client radio band support capability (2.4G/5G/both)
vHTCap	int32	snapshot	NULL	Client HT/VHT capability (non-HT/HT/ VHT)
streamCap	int32	snapshot	NULL	Client STBC capability
BTMCap	int32	snapshot	NULL	Client BTM capability
medianTxMCSRate	uint32	delta	NULL	Client median TX MCS rate in this bin
medianRxMCSRate	uint32	delta	NULL	Client median RX MCS rate in this bin
clientAuthTTC	uint64	snapshot	NULL	Client auth TTC latest value in this bin
clientAssocTTC	uint64	snapshot	NULL	Client assoc TTC latest value in this bin
clientEapTTC	uint64	snapshot	NULL	Client EAP TTC latest value in this bin
clientRadiusTTC	uint64	snapshot	NULL	Client Radius TTC latest value in this bin
clientDhcpTTC	uint64	snapshot	NULL	Client DHCP TTC latest value in this bin
roamingFailureCount	uint32	delta	SUM	Roaming faliure count for client(delta value)
roamingSuccessCount	uint32	delta	SUM	Roaming success count on client(delta value)
deviceType	int32	snapshot	NULL	Client device type, such as Laptop, Smartphones, Tablets etc.
osVendorType	int32	snapshot	NULL	Client OSVendor type, such as Windows, Android, Apple etc.
modelName	string	snapshot	NULL	Client's device model name, such as Windows 7, Mac OS X etc.
TxPER	string	delta	NULL	Client TX Packet Error Rate in this bin

TABLE 34 AP Report Bin Client Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
authMethod	string	None	None	None
clientTTC	ttcData	snapshot	NULL	Multiple client sesssion TTC data
sessDeauthTime	sessDeauthData	snapshot	NULL	Multiple client sesssion deauth time
WiFi6Cap	int32	snapshot	NULL	Client WiFi6 Capability

TABLE 35 Hccd Connection Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
timestamp	uint64	snapshot	NULL	The time for each message(packet) observed
message_id	int32	snapshot	NULL	Message identity for each packet
source	int32	snapshot	NULL	Message(packet) source module (like as client, AP, Cblade,...etc)
destination	int32	snapshot	NULL	Message(packet) destination module
status_code	int32	snapshot	NULL	Message status (success or failure)

TABLE 36 Hccd Client Connection Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
client_mac	string	snapshot	NULL	Client mac for each connection
timestamp	uint64	snapshot	NULL	The time that each connection started
connection_status	int32	snapshot	NULL	This connection session status - success or failure
failed_msg_id	int32	snapshot	NULL	The failure message identifier for this connection
hccdConnMessages	.HccdConnMessage	Serialization	NULL	Serialization data for all of message ID in this client session
wlan_id	uint32	snapshot	NULL	WLAN identifier for this client association
radio_id	uint32	snapshot	NULL	Radio identifier for this client association
ssid	string	snapshot	NULL	SSID for this client association
wlanType	string	snapshot	NULL	WLAN type used
failure_type	int32	snapshot	NULL	Category for the connection failure (auth/assoc/eap/radius/dhcp failure)
vlan	int32	snapshot	NULL	VLAN identifier value for this client
reason_code	int32	snapshot	NULL	Reason code for disconnection
info	string	snapshot	NULL	Reason for disconnect

TABLE 37 AP Report Bin Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
bin	int	snapshot	NULL	Bin data number xxx AP divides 24 hours into 96 bins, - 1 bin is 15 minutes.Bin1 time period is 00:00~00:15, Bin2 time period is 00:15~00:30 and so on. It is a number from 1 to 96.Then one can see the stats time period according to this value.
uptime_r	int	snapshot	NULL	Uptime in one report duration Example: AP boots up at 10:08, so the uptime_r is 420 seconds(10:15 - 10:08) in Bin (10:00~10:15). If AP keeps work well, and now is 11:02, then uptime_r is 900 seconds in Bin (10:45~11:00)

TABLE 38 AP Firewall Profile Stats Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
firewallProfileId	string	snapshot	NULL	Firewall profile identifier
firewallProfileName	string	snapshot	NULL	Firewall profile name
firewallProfileHitCount	uint64	delta	SUM	Firewall profile hit count
l2AclPolicyId	string	snapshot	NULL	L2 ACL policy identifier
l2AclPolicyHitCount	uint64	delta	SUM	L2 ACL policy hit count
l3AclPolicyId	string	snapshot	NULL	L3 ACL policy identifier
l3AclPolicyHitCount	uint64	delta	SUM	L3 ACL policy hit count
devicePolicyId	string	snapshot	NULL	Device policy identifier
devicePolicyHitCount	uint64	delta	SUM	Device policy hit count
avcPolicyId	string	snapshot	NULL	AVC policy identifier
avcPolicyHitCount	uint64	delta	SUM	AVC policy hit count
urlFilteringId	string	snapshot	NULL	Url filtering policy identifier
urlFilteringHitCount	uint64	delta	SUM	Url filtering policy hit count

TABLE 39 AP Report Stats Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
ap	string	snapshot	NULL	AP MAC address
timestamp	uint64	snapshot	NULL	Timestamp for this report
seqNumber	uint64	snapshot	NULL	Sequence record number for AP report.
zone_id	string	snapshot	NULL	Zone UUID
binCount	Struct APReportBin	Serialization	NULL	Total number of bin data.

TABLE 39 AP Report Stats Information (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
binClient	Struct APReportBinClient	Serialization	NULL	Total number of Client bin data.
binIPSec	Struct APReportBinIPSec	Serialization	NULL	The number of IPSec tunnel.
binTunnel	Struct APReportBinTunnel	Serialization	NULL	The number of tunnel data.
binRadio	Struct APReportBinRadio	Serialization	NULL	The number of radio data.
deviceName	string	snapshot	NULL	AP devices configured by the administrator.
apgroup_id	string	snapshot	NULL	AP Group UUID
cluster_id	string	snapshot	NULL	Cluster UUID.
domain_id	string	snapshot	NULL	Domain UUID.
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	MAP UUID
aptenant_name	string	snapshot	NULL	AP tenant name.
zone_name	string	snapshot	NULL	Zone name.
apgroup_name	string	snapshot	NULL	AP Group name
domain_name	string	snapshot	NULL	Domain name
sampleTime	uint64	snapshot	NULL	The timestamp for sampling stats data
aggregationInterval	uint32	snapshot	NULL	The interval time to aggregate stats together
apIpAddress	string	snapshot	NULL	IP address of the AP
apIpv6Address	string	snapshot	NULL	IPv6 address of the AP
hccdClientConnections	.HccdClientConnection	Serialization	NULL	Serialization data for HCCD connection data
HeartbeatLatency	float	snapshot	AVG	Average latency that is measured by heartbeat lost
PingLatency	float	snapshot	AVG	Average latency that is measured by ping
PingLossCount	uint32	snapshot	NULL	Count for ping lost
isMonitoringEnabled	BOOL	snapshot	NULL	Is a monitoring AP or not
firewallProfileStats	APFirewallProfileStats	snapshot	NULL	Firewall Profile Stats.
apRadioDeploy	string	snapshot	NULL	Indicate AP all radio band deploy
ideal_stats_required	AttributeMap			
rflowMaxCount	TYPE_UINT64	snapshot	NULL	Report max-rflow count

ap_rogue.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message ReportType {

    enum RogueType {
                DISCOVERY = 0;
                UPDATE = 1;
                DISAPPEAR = 2;
        }

    optional string rogueMac = 1;
    optional uint32 rssi = 2;
    optional string encryption = 3;
    optional string radio = 4;
    optional uint32 channel = 5;
    optional uint64 timeStamp = 6;
    optional string ssid = 7;
    optional int32 wlanId = 8;
    optional string rogueAPMac = 9;
    optional int32 isSendEvent = 10;
    optional string type = 11;
    optional uint32 prevReportChannel = 12;
    optional string prevReportType = 13;
    optional RogueType rogueType = 14;
    optional uint32 rogueTypeInfo = 15;
}

message RogueAPStats {
    required uint32 version = 1;
    optional string apMac = 2;
    optional string apName = 3;
    optional string zone_id = 4;
    optional string protect = 5;
    repeated ReportType apRogueUpdate = 6;
    optional string apgroup_id = 7;
    optional string cluster_id = 8;
    optional string domain_id = 9;
    optional string aptenant_id = 10;
    optional string map_id = 11;
    optional string aptenant_name = 12;
    optional string zone_name = 13;
    optional string apgroup_name = 14;
    optional string domain_name = 15;
    optional int32  controllerShouldFlush = 16;
    optional uint64 sampleTime = 17;
    optional uint32 aggregationInterval = 18;
    optional uint64 timestamp = 19;
    optional uint32 operation_type = 20;
    optional uint64 totPpduDur = 21;
    optional uint64 totScanTime = 22;
    optional string serialNumber = 23;
    optional string apRadioDeploy = 24;
}
```

Field Description

TABLE 40 Enum Rogue Report

Name	Value	Description
Discovery	0	Rogue type possible value
Update	1	Rogue type possible value
Disappear	2	Rogue type possible value

TABLE 41 AP Rogue Report Type Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
rogueMac	string	snapshot	NULL	Mac address of rogue AP
rssi	uint32	snapshot	NULL	peer node rssi value
encryption	string	snapshot	NULL	is encryption or not (encrypted or open)
radio	string	snapshot	NULL	peer node radio type
channel	uint32	snapshot	NULL	peer node channel used
timeStamp	uint64	snapshot	NULL	last detected time for this peer node
ssid	string	snapshot	NULL	peer node ssid
wlanId	int32	snapshot	NULL	peer node wlanId
rogueAPMac	string	snapshot	NULL	peer node mac address
isSendEvent	int32	snapshot	NULL	is event send out for this peer node
type	string	snapshot	NULL	current rouge type for this peer node
prevReportChannel	string	snapshot	NULL	previous channel is used by peer node
prevReportType	int32	snapshot	NULL	previous rogue type for this peer node
rogueType	RogueType	snapshot	NULL	this peer node rogue type status(discover, update, disappear)
rogueTypeInfo	uint32	snapshot	NULL	current rouge type for this peer node

TABLE 42 Rogue AP Stats Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
version	uint32	snapshot	NULL	Version number for this stats report
apMac	string	snapshot	NULL	AP mac address
apName	string	snapshot	NULL	AP name
zone_id	string	snapshot	NULL	Zone UUID
protect	char	snapshot	NULL	Is malicious protected or not
apRogueUpdate	.ReportType	Serialization	NULL	Serialization data for all of rouge entry
apgroup_id	string	snapshot	NULL	AP group UUID
cluster_id	string	snapshot	NULL	Cluster UUID
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID

TABLE 42 Rogue AP Stats Information (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
map_id	string	snapshot	NULL	MAP uuid
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name
apgroup_name	string	snapshot	NULL	AP group name
domain_name	string	snapshot	NULL	Domain name
controllerShouldFlush	int32	snapshot	NULL	An indication to flush all rouge stats when first report
sampleTime	uint64	snapshot	NULL	Timestamp for generating this stats report
aggregationInterval	uint32	snapshot	NULL	Interval time for aggregating data
timestamp	uint64	snapshot	NULL	Timestamp for generating this stats report
operation_type	uint32	snapshot	NULL	1: new rogue report, 2: full rogue listing report, 3: new rogue client report, 4: full rogue client listing report
totPpduDur	uint64	snapshot	SUM	Timestamp for generating this stats report
totScanTime	uint64	snapshot	SUM	Timestamp for generating this stats report
serialNumber	string	snapshot	NULL	The serial number in AP borad data.
apRadioDeploy	string	snapshot	NULL	Indicate AP all radio band deploy

ap_status.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APStatusTunnel {
    optional string gw = 1;
    optional int32 index = 2;
    optional int32 type = 3;
    optional int32 isActive = 4;
    optional uint64 cICMP = 5;
    optional uint64 cNonICMP = 6;
    optional uint64 cDisconnect = 7;
    optional uint64 rxBytes = 8;
    optional uint64 rxPkts = 9;
    optional uint64 rxDropPkts = 10;
    optional uint64 rxErrPkts = 11;
    optional uint64 txBytes = 12;
    optional uint64 txPkts = 13;
    optional uint64 txDropPkts = 14;
    optional uint64 txErrPkts = 15;
    optional uint64 txFragPkts = 16;
    optional string uptime = 17;
    optional uint64 reEstablishment = 18;
    optional uint64 kaRetryCnt = 19;
    optional uint64 kaSentCnt = 20;
    optional uint64 kaLostCnt = 21;
    optional string reason = 22;
    optional string suggest = 23;
    optional string dpKey = 24;
}

message APStatusIPSecStats {
    enum IPsecTunnelType {
        RUCKUSGRE = 0;
        SOFTGRE = 1;
    }
    optional uint64 ipsecSessionTime = 1;
    optional uint64 ipsecTxPkts = 2;
    optional uint64 ipsecRxPkts = 3;
    optional uint64 ipsecTxBytes = 4;
    optional uint64 ipsecRxBytes = 5;
    optional uint64 ipsecTxDropPkts = 6;
    optional uint64 ipsecRxDropPkts = 7;
    optional uint64 ipsecTxIdleTime = 8;
    optional uint64 ipsecRxIdleTime = 9;
    optional IPsecTunnelType ipsecTunnelType = 10;
}

message APStatusWlan {
    optional string    ssid = 1;
    optional string    bssid = 2;
    optional string    ap = 3;
    optional string    radioMode = 4;
    optional string    ap80211RadioMode = 5;
    optional int32    channel = 6;
    optional uint64    rxBytes = 7;
    optional uint64    txBytes = 8;
    optional uint64    rxFrames = 9;
    optional uint64    txFrames = 10;
    optional uint64    txBcastFrames = 11;
    optional uint64    txMcastFrames = 12;
    optional uint64    txDataFrames = 13;
    optional uint64    txDataBytes = 14;
```

```
optional uint64    txMgmtFrames = 15;
optional uint64    txMgmtBytes = 16;
optional uint64    txDropDataFrames = 17;
optional uint64    txDropMgmtFrames = 18;
optional uint64    rxBcastFrames = 19;
optional uint64    rxMcastFrames = 20;
optional uint64    rxDataFrames = 21;
optional uint64    rxDataBytes = 22;
optional uint64    rxMgmtFrames = 23;
optional uint64    rxMgmtBytes = 24;
optional int32    totalNumClients = 25;
optional int32    vlan = 26;
optional int32    wsgWlanId = 27;
optional int32    wlanId = 28;
optional string wlangroup_name = 29;
optional string wlangroup_id = 30;
optional string wlantenant_id = 31;
optional string wlantenant_name = 32;
optional uint32 ftassoc_success = 33;
optional uint32 ftassoc_failure = 34;
optional uint32 is_probe_withheld = 35;
optional uint32 is_auth_withheld = 36;
optional uint32 beacon_queued = 37;
optional uint32 beacon_reaped = 38;
/* jump to index 1000, for new requirement from SNMP and SCI */
optional string WlanName = 1001;
optional string AuthMethod = 1002;
optional string EncryptMethod = 1003;
optional uint32    IsGuest = 1004;
optional uint32    IsBcastDisable = 1005;
optional uint32    UpRateLimit = 1006;
optional uint32    DownRateLimit = 1007;
optional uint32    IsTunnel = 1008;
optional uint32    RxByteRate = 1009;
optional uint32    TxByteRate = 1010;
optional uint64    RxDropDataBytes = 1011;
optional uint64    TxDropDataBytes = 1012;
optional uint64    RxDropDataPkts = 1013;
optional uint64    TxDropDataPkts = 1014;
optional uint64    TxRetryBytes = 1015;
optional uint64    TxRetryPkts = 1016;
optional uint64    RxErrorPkts = 1017;
optional uint64    TxErrorPkts = 1018;
optional uint32    RxPktErrorRate = 1019;
optional uint32    TxPktErrorRate = 1020;
optional uint32    NumAuthClients = 1021;
optional uint32    NumAssocReq = 1022;
optional uint32    NumAssocResp = 1023;
optional uint32    NumReassocReq = 1024;
optional uint32    NumReassocResp = 1025;
optional uint32    NumAssocFail = 1026;
optional uint32    NumAssocDeny = 1027;
optional uint32    DisssocAbnormal = 1028;
optional uint32    NumDisssocCapacity = 1029;
optional uint32    MumDisssocLeave = 1030;
optional uint32    NumDisssocMisc = 1031;
optional uint32    AssocSuccessRate = 1032;
optional uint32    AssocFailRate = 1033;
optional uint32    NumAuthReq = 1034;
optional uint32    NumAuthResp = 1035;
optional uint32    NumAuthSuccess = 1036;
optional uint32    NumAuthFail = 1037;
optional uint32    AuthFailRate = 1038;
optional uint32    RtsThreshold = 1039;
optional uint32 lookup_success_fst = 1040;
optional uint32 lookup_success_sessmgr = 1041;
optional uint32 roaming_success = 1042;
optional uint32 connection_attempts = 1043;
optional uint32 pmkid_success = 1044;
optional uint32 pmkid_mismatch = 1045;
optional uint32 pmkid_missing = 1046;
optional uint32 ft_auth_success = 1047;
```

```
    optional uint32 ftie_mismatch = 1048;
    optional uint32 ftie_missing = 1049;
}

message APStatusRadio {
    optional int32    radioId = 1;
    optional int32    channel = 2;
    optional string    mode = 3;
    optional string    band = 4;
    optional string    radioMode = 5;
    optional string    txPower = 6;
    optional uint32    phyError = 7;
    optional string    channelBlacklist = 8;
    optional int32    noiseFloor = 9;
    optional uint64    rxBytes = 10;
    optional uint64    rxFrames = 11;
    optional uint64    rxRadioBytes = 12;
    optional uint64    rxRadioFrames = 13;
    optional uint64    txBytes = 14;
    optional uint64    txFrames = 15;
    optional uint64    txRadioBytes = 16;
    optional uint64    txRadioFrames = 17;
    optional uint64    retry = 18;
    optional uint32    drop = 19;
    optional uint64    rxMulticast = 20;
    optional uint64    txMulticast = 21;
    optional uint32    total = 22;
    optional uint32    busy = 23;
    optional uint32    rx = 24;
    optional uint32    tx = 25;
    optional uint32    channelWidth = 26;
    repeated APStatusWlan wlans = 27;
    optional string    ap = 28;
    optional uint32   latency = 29;
    optional uint32   capacity = 30;
    optional float connectionFailure = 31;
    optional uint32   connectionAuthFailureCount = 32;
    optional uint32   connectionAssocFailureCount = 33;
    optional uint32   connectionTotalCount = 34;
    optional uint32   numOfChannelChange = 35;
    optional bool isLatencyFlagged = 36;
    optional bool isCapacityFlagged = 37;
    optional bool isConnectionFailureFlagged = 38;
    optional bool isAirtimeFlagged = 39;
    optional bool isRadioEnabled = 40;
    optional uint32   secondaryChannel = 41;
    optional int32    eirp = 42;
    optional uint32 connectionTotalFailureCount = 43;
    optional uint32 tx_rts_cnt = 44;
    optional uint32 totalFailureClientCount = 45;
    optional uint32 totalClientCnts = 46;
    optional string    chainmask = 47;
    optional uint64    rxMcastBytes = 48;
    optional uint64    txMcastBytes = 49;
    optional uint64    rxMcastDataBytes = 50;
    optional uint64    txMcastDataBytes = 51;
    optional int32    channelWidthGroup = 52;
    /* jump to index 1000, for new requirement from SNMP and SCI */
    optional int32 PowerMgmtEnable = 1001;
    optional int32 MeshEnable = 1002;
    optional uint64 RxErrorPkts = 1003;
    optional uint64 TxErrorPkts = 1004;
    optional uint32 RxPktErrorRate = 1005;
    optional uint32 TxPktErrorRate = 1006;
    optional uint32 TxPktRetryRate = 1007;
    optional uint64 TxRetryBytes = 1008;
    optional uint64 RxDropBytes = 1009;
    optional uint64 TxDropBytes = 1010;
    optional uint64 RxDropPkts = 1011;
    optional uint64 TotalAssocTime = 1012;
    optional uint32 NumAuthClients = 1013;
    optional uint32 NumMaxClients = 1014;
```

```
    optional uint32 NumAuthReqs = 1015;
    optional uint32 NumAuthResps = 1016;
    optional uint32 NumAuthSuccess = 1017;
    optional uint32 NumAuthFail = 1018;
    optional uint32 AuthFailRate = 1019;
    optional uint32 NumAssocReq = 1020;
    optional uint32 NumAssocResp = 1021;
    optional uint32 NumReassocReq = 1022;
    optional uint32 NumReassocResp = 1023;
    optional uint32 NumAssocSuccess = 1024;
    optional uint32 NumAssocFail = 1025;
    optional uint32 NumAssocDeny = 1026;
    optional uint32 AssocSuccessRate = 1027;
    optional uint32 AssocFailRate = 1028;
    optional uint32 ResourceUtil = 1029;
    optional uint64 RxSignalPkts = 1030;
    optional uint64 TxSignalPkts = 1031;
    optional uint64 TotalSignalPkts = 1032;
    optional uint32 AntennaGain = 1033;
    optional uint32 BeaconPeriod = 1034;
    optional uint32 RtsThreshold = 1035;
    optional uint32 FragThreshold = 1036;
    optional uint32 RxWepFail = 1037;
    optional uint32 RxDecryptCrcError = 1038;
    optional uint32 RxMicError = 1039;
    optional uint32 Rssi = 1040;
    optional uint32 TxPowerOffset = 1041;
    optional uint32 RxDesense = 1042;
}

message APStatusBrownout {
    optional uint64    timestamp = 1;
    optional int32    events = 2;
    optional int32    pwrType = 3;
}

message APStatusIPSec {
    optional string    ipsecActiveServerIP = 1;
    optional string    ipsecVirtualIPv4 = 2;
    optional string    ipsecVirtualIPv6 = 3;
    optional string    ipsecEffectiveIKESA = 4;
    optional string    ipsecEffectiveESPSA = 5;
}

message APStatusSystem {
    enum APState {
        Online = 1;
        Flagged = 2;
    }
    optional string    ap = 1;
    optional string    usbDeviceVersion = 2;
    optional string    usbDeviceVID = 3;
    optional string    usbDevicePID = 4;
    optional string    gpsInfo = 5;
    optional string    countryCode = 6;
    optional uint64    timestamp = 7;
    optional uint64    seqNumber = 8;
    optional string    zone_id = 9;
    optional string    zoneName = 10;
    optional string    timeZone = 11;
    optional string    gatewayIp = 12;
    optional string    lastRebootReason = 13;
    optional uint32    totalBootCount = 14;
    optional uint32    mtuSize = 15;
    optional uint32    rejoinCount = 16;
    optional string    rejoinReason = 17;
    optional string    oops = 18;
    optional uint32    lossConnectBootCnt = 19;
    optional string    deviceName = 20;
    optional string    location = 21;
    optional string    fwVersion = 22;
    optional int32    devSupportUsb = 23;
```

```
optional int32    deviceIpMode = 24;
optional string    ip = 25;
optional string    ipv6 = 26;
optional string    ipsecIp = 27;
optional string    apConnectedIp = 28;
optional int32    uptime = 29;
optional string    mountState = 30;
optional int32    currentTemperature = 31;
optional int32    lifeMaxTemperature = 32;
optional int32    lifeMinTemperature = 33;
optional string dnatInfo = 34;
optional string rksDpIp = 35;
optional string rksDpIpOnly = 36;
optional string ipType = 37;
optional uint32 isIpTypeChanged = 38;
optional uint32 managementVlan = 39;
optional APState apState = 40;
optional bool isConnectionTotalCountFlagged = 41;
optional uint32 totalConnectedClient = 42;
optional uint32 crashDump = 43;
optional string altitudeUnit = 44;
optional uint32 altitudeValue = 45;
optional uint32 poeMode = 46;
optional uint32 poeModeSetting = 47;
optional string ipv6Type = 48;
optional double freeMemoryPercentage = 49;
optional double freeStoragePercentage = 50;
optional uint32 poeUnderPowered = 51;
optional string chainmask5G = 52;
optional string chainmask24G = 53;
optional bool isIoTEnable = 54;
optional bool isUSBEnable = 55;
optional bool isPoEOutEnable = 56;
optional bool isSecEthEnable = 57;
optional string config_id = 58;
optional string upd_config_id = 59;
optional ConfigFwStatus config_status = 60;
optional string fw_upd_version = 61;
optional ConfigFwStatus fw_upd_status = 62;
optional string fw_upd_failed_reason = 63;
optional string fw_request_id = 64;
optional string config_failed_feature_name = 65;
optional string config_failed_feature_attribute = 66;
optional string config_failed_err_message = 67;
optional string config_failed_feature_attr_value = 68;
optional string config_request_id = 69;
optional int32 is_secure_boot_enable = 70;

/* jump to index 1000, for new requirement from SNMP and SCI */
optional double cpuPercentage = 1001;
optional uint64 totalMemory = 1002;
optional uint64 freeMemory = 1003;
optional string model = 1004;
optional string serialNumber = 1005;
optional string desc = 1006;
optional int32 numRadio = 1007;
optional string szConnCpIp = 1008;
optional string szConnCpIpv6 = 1009;
optional string szConnDpIp = 1010;
optional string szConnDpIpv6 = 1011;
optional string netmask = 1012;
optional string IpDnsSvr1 = 1013;
optional string IpDnsSvr2 = 1014;
optional string Ipv6DnsSvr1 = 1015;
optional string Ipv6DnsSvr2 = 1016;
optional int32 ApStatus = 1017;
optional uint64 firstJoinTime = 1018;
optional uint64 lastBootTime = 1019;
optional uint64 lastConfSyncTime = 1020;
optional uint64 freeStorage = 1021;
optional int32 ethPortStatus = 1022;
optional int32 ethStateChange = 1023;
```

```
    optional uint32 numRogues = 1024;
    optional uint32 numAuthClients = 1025;
    optional uint32 rxByteRate = 1026;
    optional uint32 txByteRate = 1027;
    optional uint64 rxErrorPkts = 1028;
    optional uint64 txErrorPkts = 1029;
    optional uint64 RxDropPkts = 1030;
    optional uint64 LanStatsRxBytes = 1031;
    optional uint64 LanStatsTxBytes = 1032;
    optional uint64 LanStatsRxPkts = 1033;
    optional uint64 LanStatsTxPkts = 1034;
    optional uint64 LanStatsRxErrorPkts = 1035;
    optional uint64 LanStatsTxErrorPkts = 1036;
    optional uint64 LanStatsRxBcastPkts = 1037;
    optional uint64 LanStatsTxBcastPkts = 1038;
    optional uint64 LanStatsRxMcastPkts = 1039;
    optional uint64 LanStatsTxMcastPkts = 1040;
    optional uint64 LanStatsRxUcastPkts = 1041;
    optional uint64 LanStatsTxUcastPkts = 1042;
    optional uint64 LanStatsRxDroppedPkts = 1043;
    optional uint64 LanStatsTxDroppedPkts = 1044;
    optional uint64 LanStatsRxByteRate = 1045;
    optional uint64 LanStatsTxByteRate = 1046;
    optional uint64    TxDropPkts = 1047;
}

message LanPortStatus {
    optional uint32 port = 1;
    optional string interface = 2;
    optional string dot1x = 3;
    optional string logicLink = 4;
    optional string phyLink = 5;
    optional string sfpInfo = 6;
    optional string wanConnectivity = 7;
    optional string phyCapability = 8;
}

message CableModemInfo {
    optional string cmMac = 1;
    optional string cmIp = 2;
    optional string cmFwVersion = 3;
    optional uint64    cmUptime = 4;
    optional string cmSerialNumber = 5;
    optional string cmIpv6 = 6;
    optional string cmCapabilities = 7;
    optional uint32 cmRangingTimeout = 8;
    optional uint32 cmStatusValue = 9;
        optional string cmStatusCode = 10;
        optional string cmStatusTxPower = 11;
        optional uint32 cmStatusResets = 12;
        optional uint32 cmStatusLostSyncs = 13;
        optional uint32 cmStatusInvalidMaps = 14;
        optional uint32 cmStatusInvalidUcds = 15;
        optional uint32 cmStatusInvalidRangingResponses = 16;
        optional uint32 cmStatusInvalidRegistrationResponses = 17;
        optional uint32 cmStatusT1Timeouts = 18;
        optional uint32 cmStatusT2Timeouts = 19;
        optional uint32 cmStatusT3Timeouts = 20;
        optional uint32 cmStatusT4Timeouts = 21;
        optional uint32 cmStatusRangingAborteds = 22;
        optional uint32 cmStatusDocsisOperMode = 23;
        optional uint32 cmStatusModulationType = 24;
        optional string cmStatusEqualizationData =25;
}

message APStatusLBS {
    optional bool isLBSEnable = 1;
    optional bool isLBSConnected = 2;
    optional bool isSupportLBS = 3;
}

message CellularInfo {
```

```
        optional string cellularWanInterface = 1; // "wwan0"
        optional string cellularConnectionStatus = 2; // "2G" / "3G" / "4G" / "Not Connected"
        optional string cellularIMSISIM0 = 3;
        optional string cellularIMSISIM1 = 4;
        optional string cellularICCIDSIM0 = 5;
        optional string cellularICCIDSIM1 = 6;
        optional string cellularIsSIM0Present = 7; // YES or NO
        optional string cellularIsSIM1Present = 8; // YES or NO
        optional uint64 cellularTxBytesSIM0 = 9;
        optional uint64 cellularTxBytesSIM1 = 10;
        optional uint64 cellularRxBytesSIM0 = 11;
        optional uint64 cellularRxBytesSIM1 = 12;
        optional string cellularActiveSim = 13; // "SIM0" / "SIM1"
        optional string cellularIPaddress = 14;
        optional string cellularSubnetMask = 15;
        optional string cellularDefaultGateway = 16;
        optional string cellularOperator = 17;
        optional int32 cellular3G4GChannel = 18;
        optional string cellularSignalStrength = 19;
        optional string cellularCountry = 20;
        optional int32 cellularRadioUptime = 21;
        optional string cellularLTEFirmware = 22;
        optional int64 cellularSwitchCountSIM0 = 23;
        optional int64 cellularSwitchCountSIM1 = 24;
        optional int64 cellularNWLostCountSIM0 = 25;
        optional int64 cellularNWLostCountSIM1 = 26;
        optional int64 cellularCardRemovalCountSIM0 = 27;
        optional int64 cellularCardRemovalCountSIM1 = 28;
        optional int64 cellularDHCPTimeoutCountSIM0 = 29;
        optional int64 cellularDHCPTimeoutCountSIM1 = 30;
        optional string cellularRoamingStatus = 31;
        optional string cellularIMEI = 32;
    optional int32 cellularRSRP = 33;
    optional int32 cellularRSRQ = 34;
    optional int32 cellularSINR = 35;
    optional int32 cellularRSCP = 36;
    optional int32 cellularECIO = 37;
    optional string cellularBand = 38;
    optional string cellularUplinkBandwidth = 39;
    optional string cellularDownlinkBandwidth = 40;
    repeated GpsHistoryData gpsHistory = 41;
}

message GpsHistoryData {
     optional uint64 timestamp = 1;
     optional string latitude = 2;
     optional string longitude = 3;
}

message APStatusData {
    optional APStatusSystem APSystem = 1;
    optional APStatusIPSec APIPSec = 2;
    repeated APStatusBrownout APBrownout = 3;
    repeated APStatusRadio APRadio = 4;
    optional APStatusIPSecStats APIPSecStats = 5;
    repeated APStatusTunnel APTunnel = 6;
    repeated LanPortStatus lanPortStatus = 7;
    optional CableModemInfo cableModemInfo = 9;
    optional APStatusLBS APStatusLBS = 10;
    optional CellularInfo cellularInfo = 11;
    optional DeviceSessionCacheStats deviceSessionCacheStats = 12;
    optional AFCInfo afcInfo = 13;
    optional APStatusVxLAN vxlanStatus = 14;
}

message APStatus {
    required uint32 version = 1;
    optional APStatusData ap_status_data = 2;
    optional string zone_id = 3;
    optional string apgroup_id = 4;
    optional string cluster_id = 5;
    optional string domain_id = 6;
```

```
    optional string aptenant_id = 7;
    optional string map_id = 8;
    optional string aptenant_name = 9;
    optional string zone_name = 10;
    optional string apgroup_name = 11;
    optional string domain_name = 12;
    optional string wlangroup24G_id = 13;
    optional string wlangroup24G_name = 14;
    optional string wlangroup5G_id = 15;
    optional string wlangroup5G_name = 16;
    optional uint64 sampleTime = 17;
    optional uint32 aggregationInterval = 18;
    optional string map_name = 19;
    optional string apMac = 20;
    optional int32 lacp_state = 21;
    repeated ApStatusWlanGroup wlan_groups = 22;
    optional string apRadioDeploy = 23;
}
```

Field Description

TABLE 43 AP Status Tunnel Information

Name	Value	Description
Online	1	AP state possible value
Flagged	2	AP state possible value

TABLE 44 AP Status Tunnel Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
gw	string	snapshot	NULL	Application name
index	int32	snapshot	NULL	Current tunnel index
type	int32	snapshot	NULL	Tunnel type
isActive	int32	snapshot	NULL	Current tunnel state (active or inactive)
cICMP	uint64	snapshot	MAX	Number of keepalive packets sent
cNonICMP	uint64	snapshot	MAX	Number of keepalive packets lost
cDisconnect	uint64	snapshot	MAX	Number of SoftGRE sessions terminated
rxBytes	uint64	snapshot	SUM	Total bytes received
rxPkts	uint64	snapshot	SUM	Total packets received
rxDropPkts	uint64	snapshot	SUM	No space in linux buffers. The AP received a frame completely, and AP has to allocate a memory for this frame. Then pass to next handler who interested to this frame. The AP drop the frame if it has no enough memory. The counter plus one if this case happen.
rxErrPkts	uint64	snapshot	SUM	Number of bad packets received
txBytes	uint64	snapshot	SUM	Total bytes transmitted
txPkts	uint64	snapshot	SUM	Total packets transmitted

TABLE 44 AP Status Tunnel Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
txDropPkts	uint64	snapshot	SUM	No space available in linux AP has to allocate a memory to store the packet that AP plans to transmit to client. The AP may drop it if AP has no enough memory. Then the counter plus one.
txErrPkts	uint64	snapshot	SUM	Packet transmit problems. The AP plans to transmit packet to client. But somehow AP can't transmit to client successfully, and AP drops the packets at the end. Then the counter plus one.
txFragPkts	uint64	snapshot	SUM	Total fragmented Tx packets
uptime	string	snapshot	NULL	Tunnel uptime
reEstablishment	uint64	snapshot	NULL	Number of tunnel reestablishment
kaRetryCnt	uint64	snapshot	NULL	Keep alive retry count
kaSentCnt	uint64	snapshot	NULL	Number of keep alive sent
kaLostCnt	uint64	snapshot	NULL	Number of keep alive lost
reason	string	snapshot	NULL	Reason for last re-connection
suggest	string	snapshot	NULL	Reason for last re-connection
dpKey	string	snapshot	NULL	DP key

TABLE 45 AP Status IPSec Statistics

Name	Value	Description
RUCKUSGRE	0	Tunnel type possible value
SOFTGRE	1	Tunnel type possible value

TABLE 46 AP Status IPSec Statistics

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
ipsecSessionTime	uint64	snapshot	MAX	Session time
ipsecTxPkts	uint64	snapshot	MAX	Total packets transmitted
ipsecRxPkts	uint64	snapshot	MAX	Total packets received
ipsecTxBytes	uint64	snapshot	MAX	Total bytes transmitted
ipsecRxBytes	uint64	snapshot	MAX	Total bytes received
ipsecTxDropPkts	uint64	snapshot	MAX	Total Tx packets dropped
ipsecRxDropPkts	uint64	snapshot	MAX	Total Rx packets dropped
ipsecTxIdleTime	uint64	snapshot	MAX	Tx idle time
ipsecRxIdleTime	uint64	snapshot	MAX	Rx idle time
ipsecTunnelType	.APStatusIPSecStats.I PsecTunnelType	snapshot	NULL	Tunnel type

TABLE 47 AP Status WLAN Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ssid	string	snapshot	NULL	SSID of the WLAN
bssid	string	snapshot	NULL	BSSID of the WLAN
ap	string	snapshot	NULL	MAC address of the AP
80211RadioMode	string	snapshot	NULL	Radio mode used by the AP
ap80211RadioMode	string	snapshot	NULL	Radio mode used by the AP
channel	int	snapshot	NULL	Radio channel used by the AP
rxBytes	uint64	delta	SUM	Total data and management packet data size received
txBytes	uint64	delta	SUM	Total data and management packet data size transmitted
rxFrames	uint64	delta	SUM	Total number of data and management packets received
txFrames	uint64	delta	SUM	Total number of data and management packets transmitted
txBcastFrames	uint64	snapshot	SUM	Total broadcast packets transmitted
txMcastFrames	uint64	snapshot	SUM	Total multicast packets transmitted
txDataFrames	uint64	snapshot	SUM	Accumulated number of packets transmitted
txDataBytes	uint64	snapshot	SUM	Accumulated data bytes transmitted
txMgmtFrames	uint64	snapshot	SUM	Accumulated number of packet transmitted
txMgmtBytes	uint64	snapshot	SUM	Accumulated management bytes transmitted
txDropDataFrames	uint64	snapshot	SUM	Total Tx data frames dropped (or dropped by MQ) . In AP internal design, it has a messages queue(MQ) to queue all packets that AP plans to forward to clients. Then AP transmit packets to clients accoring the priority / scheduling/... The MQ is Ruckus propritary internal design.
txDropMgmtFrames	uint64	snapshot	SUM	Total Tx management frames dropped (or dropped by MQ) . In AP internal design, it has a messages queue(MQ) to queue all packets that AP plans to forward to clients. Then AP transmit packets to clients accoring the priority / scheduling/... The MQ is Ruckus propritary internal design.
rxBcastFrames	uint64	snapshot	SUM	Total broadcast packets received
rxMcastFrames	uint64	snapshot	SUM	Total multicast packets received
rxDataFrames	uint64	snapshot	SUM	Accumulated number of packets received
rxDataBytes	uint64	snapshot	SUM	Accumulate data bytes received
rxMgmtFrames	uint64	snapshot	SUM	Accumulated number of packets received

TABLE 47 AP Status WLAN Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
rxMgmtBytes	uint64	snapshot	SUM	Accumulate management bytes received
totalNumClients	int	snapshot	NULL	Current client count
vlan	int	snapshot	NULL	VLAN ID of the WLAN
wsgWlanId	int	snapshot	NULL	WLAN ID assigned by the controller
wlanId	int	snapshot	NULL	WLAN interface ID
wlangroup_name	string	snapshot	NULL	WLAN Group name
wlangroup_id	string	snapshot	NULL	WLAN Group identifier
wlantenant_id	string	snapshot	NULL	WLAN tenant identifier
wlantenant_name	string	snapshot	NULL	WLAN tenant name
ftassoc_success	uint32	snapshot	NULL	802.11r fast roaming status success
ftassoc_failure	uint32	snapshot	NULL	802.11r fast roaming status failure
is_probe_withheld	uint32	snapshot	NULL	Probe resp withheld if this is a Probe Request, Client Load Balancing and Band Steering decide whether to withhold the response).
is_auth_withheld	uint32	snapshot	NULL	Auth resp withheld (If this is an Auth Request, Client Load Balancing and Band Steering decide whether to withhold the response).
beacon_queued	uint32	snapshot	NULL	Beacon queued value from driver
beacon_reaped	uint32	snapshot	NULL	Beacon reaped value from driver
WlanName	string	snapshot	NULL	Wlan SSID
AuthMethod	string	snapshot	NULL	Wlan authentication method
EncryptMethod	string	snapshot	NULL	Wlan encryption method
IsGuest	uint32	snapshot	NULL	An indication for guest access allowed
IsBcastDisable	uint32	snapshot	NULL	Hidden SSID wlan indication
UpRateLimit	uint32	snapshot	NULL	Wlan rate limiting for upstream
DownRateLimit	uint32	snapshot	NULL	Wlan rate limiting for downstream
IsTunnel	uint32	snapshot	NULL	Tunnel wlan indication
NumAssocReq	uint32	snapshot	NULL	Assoc req count on Wlan
NumAssocResp	uint32	snapshot	NULL	Assoc resp count on Wlan
NumReassocReq	uint32	snapshot	NULL	Re-assoc req count on Wlan
NumReassocResp	uint32	snapshot	NULL	Re-assoc resp count on Wlan
NumAssocFail	uint32	snapshot	NULL	Assoc failure count on Wlan
NumAuthReq	uint32	snapshot	NULL	auth req count on Wlan
NumAuthResp	uint32	snapshot	NULL	auth resp count on Wlan
NumAuthSuccess	uint32	snapshot	NULL	success auth count on Wlan
NumAuthFail	uint32	snapshot	NULL	failure auth count on Wlan
AuthFailRate	uint32	snapshot	NULL	auth failure rate on Wlan
RtsThreshold	uint32	snapshot	NULL	Wlan RTS threshold value

TABLE 48 AP Status Radio Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
radioId	int	snapshot	NULL	Radio ID number used by the AP
Channel	int	snapshot	CONF	Radio channel used by the AP
mode	string	snapshot	NULL	Radio mode used by the AP
band	string	snapshot	NULL	Radio band used by the AP
80211RadioMode	string	snapshot	NULL	Radio mode used by the AP. Radio is 11bgn, means radio support 802.11B, 802.11G, and 802.11n. It has three capabilities. The 11bgn is not a good format if machine wants to parse. Because we have 11AC now. So we use "/" to separate each capability and then the machine can parse the string easily.
txPower	string	snapshot	NULL	Tx power of the WiFi interface
phyError	uint32	snapshot	SUM	Accumulated number of Rx phy errors
channelBlacklist	string	snapshot	NULL	Channel blacklist
noiseFloor	int	snapshot	AVG	Last recorded noise floor
rxBytes	uint64	Delta	SUM	Total data bytes received on radio
rxFrames	uint64	Delta	SUM	Total data frames received on radio
rxRadioBytes	uint64	snapshot	SUM	Total data bytes received on radio
rxRadioFrames	uint64	snapshot	SUM	Number of fragmented frames received
txBytes	uint64	Delta	SUM	Total data bytes transmitted on radio
txFrames	uint64	Delta	SUM	Total data frames transmitted on radio
txRadioBytes	uint64	snapshot	SUM	Total data bytes received on radio
txRadioFrames	uint64	snapshot	SUM	Number of fragments transmitted
retry	uint64	snapshot	SUM	Number of transmission retries
drop	uint32	snapshot	SUM	Number of excessive transmission retries
rxMulticast	uint64	snapshot	SUM	Number of multicast packets received
txMulticast	uint64	snapshot	SUM	Number of multicast packets transmitted
total	uint32	snapshot	AVG	Exponential average of total channel utilization
busy	uint32	snapshot	AVG	Exponential average of channel busy time
rx	uint32	snapshot	AVG	Exponential average of channel availability for receiving
tx	uint32	snapshot	AVG	Exponential average of channel availability for for transmitting
channelWidth	uint32	snapshot	NULL	Channel width
wlans	.APStatusWlan	Serialization	Null	Serialization data for all of wlans
ap	string	snapshot	NULL	AP mac address

TABLE 48 AP Status Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
latency	uint32	snapshot	NULL	The time taken by a packet from ethernet ingress to Radio egress or Tx complete
capacity	uint32	snapshot	NULL	The saturated throughput estimate of a link
connectionFailure	float	snapshot	NULL	The rate of client connection failure
connectionAuthFailureCount	uint32	snapshot	SUM	802.11 auth failure count on radio
connectionAssocFailureCount	uint32	snapshot	SUM	802.11 assoc failure count on radio
connectionTotalCount	uint32	snapshot	SUM	Total connection count on radio
numOfChannelChange	uint32	snapshot	SUM	Number of channel change on radio
isLatencyFlagged	bool	snapshot	SUM	Does the radio latency value exceed criteria or not
isCapacityFlagged	bool	snapshot	SUM	Does radio capacity value exceed criteria or not
isConnectionFailureFlagged	bool	snapshot	SUM	Does radio connection failure rate exceed criteria or not
isAirtimeFlagged	bool	snapshot	SUM	Does radio airtime utilization total value exceed criteria or not
isRadioEnabled	bool	snapshot	SUM	Is wifi interface up or not
secondaryChannel	uint32	snapshot	SUM	Second channel value for 80_80MHz channel width
eirp	int32	snapshot	SUM	Radio eirp value = tx_power+antenna gain
connectionTotalFailureCount	uint32	snapshot	SUM	Total connection failure count on radio
tx_rts_cnt	uint32	snapshot	NULL	TX RTS frame count
totalFailureClientCount	uint32	snapshot	SUM	Total failure client count
totalClientCnts	uint32	snapshot	NULL	Current client count per radio
chainmask	string	snapshot	NULL	AP Radio Chainmask
rxMcastBytes	unit64	snapshot	NULL	Radio rx multicast traffic byte
txMcastBytes	unit64	snapshot	NULL	Radio tx multicast traffic byte
rxMcastDataBytes	unit64	snapshot	NULL	Radio multicast rx data byte
txMcastDataBytes	unit64	snapshot	NULL	Radio multicast tx data byte
channelWidthGroup	unit64	snapshot	NULL	Radio Channelwidth group used by the AP
PowerMgmtEnable	int32	snapshot	NULL	TX power control is allowed
MeshEnable	int32	snapshot	NULL	Mesh is enabled on radio
RxErrorPkts	uint64	snapshot	NULL	RX error packet on radio
TxErrorPkts	uint64	snapshot	NULL	TX error packet on radio
RxPktErrorRate	uint32	snapshot	NULL	RX packet error rate on radio
TxPktErrorRate	uint32	snapshot	NULL	TX packet error rate on radio
TxPktRetryRate	uint32	snapshot	NULL	TX packet retry rate on radio
TxRetryBytes	uint64	snapshot	NULL	TX retry packet data bytes on radio

TABLE 48 AP Status Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
RxDropPkts	uint64	snapshot	NULL	RX drop packet count
AssocSuccessRate	uint32	snapshot	NULL	Assoc success rate on radio
AssocFailRate	uint32	snapshot	NULL	Assoc failure rate on radio
AntennaGain	uint32	snapshot	NULL	Antenna gain value on radio
BeaconPeriod	uint32	snapshot	NULL	Time period for beaon
TxPowerOffset	uint32	snapshot	NULL	Radio auto cell sizing Tx power offset (dB)
RxDesense	uint32	snapshot	NULL	Radio auto cell sizingRx Desense (dB)

TABLE 49 AP Status Brownout Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
events	int32	snapshot	NULL	Brown out event. It could be "Brownout Occurred" or "Brownout restored"
pwrType	int32	snapshot	NULL	Brown out power. It could be "PoE" or "12VDC power supply"
timeStamp	int32	snapshot	NULL	Date and time of the brownout event

TABLE 50 AP Status IPSec Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ipsecActiveServerIP	string	snapshot	NULL	IPSec remote server IP address (only populated when IPSec is enabled)
ipsecVirtualIPv4	string	snapshot	NULL	IPSec virtual IPv4 address (only populated when IPSec is enabled)
ipsecVirtualIPv6	string	snapshot	NULL	IPSec virtual IPv6 address (only populated when IPSec is enabled)
ipsecEffectiveIKESA	string	snapshot	NULL	IPSec IKE SA (only populated when IPSec is enabled)
ipsecEffectiveESPSA	string	snapshot	NULL	IPSec child SA (only populated when IPSec is enabled)

TABLE 51 AP Status System Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ap	string	snapshot	NULL	MAC address of the AP
usbDeviceVersion	string	snapshot	NULL	USB device version
usbDeviceVID	string	snapshot	NULL	USB device VLAN ID
usbDevicePID	string	snapshot	NULL	USB device product ID (PID)
gpsInfo	string	snapshot	NULL	GPS information

TABLE 51 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
countryCode	string	snapshot	NULL	Country code
seqNumber	uint64	snapshot	NULL	Counter for generating mesh statistics. The count increases by one whenever the AP generates mesh statistics. When the AP restarts, the counter also resets to zero (0).
zoneUUID	string	snapshot	NULL	Unique zone ID (for example, b381206b-2e5d-43dc-b249-e36ffae9855c) assigned by the controller
zoneName	string	snapshot	NULL	Zone name assigned by the controller. The admin configures the Zone name via controller's user interface. The controller passes it to the AP, which the zone name.
timeZone	string	snapshot	NULL	Time zone. The admin configure the time zone via SCG UI. Then SCG pass the time zone to APs.
gatewayIp	string	snapshot	NULL	Default gateway IP address of the AP
lastRebootReason	string	snapshot	NULL	Reason the AP was last rebooted
totalBootCount	uint32	snapshot	NULL	Total number of reboots since the AP was last power cycled
mtuSize	uint32	snapshot	NULL	AP br0 mtu setting. MTU stands for Maximum transmission unit. The admin coould confiure the size of MTU via SCG UI. Then SCG pass configuration to APs. So the max pake size is 1400 bytes if admin configure configure the MTU to 1400. The more details, see https://en.wikipedia.org/wiki/Maximum_transmission_unit.
rejoinCount	uint32	snapshot	NULL	Number of times the AP rejoined the controller
rejoinReason	string	snapshot	NULL	Reason the AP rejoined the controller
oops	string	snapshot	NULL	Kernel oops if there is kernel panic. The AP logs crash point when AP's kernel panic happened. Then AP report the crash point to SCG after AP boots up again.
lossConnectBootCnt	uint32	snapshot	NULL	Number of times the AP lost connection with the controller
deviceName	string	snapshot	NULL	Name of the AP
location	string	snapshot	NULL	Location of the AP
fwVersion	string	snapshot	NULL	Version of firmware installed on the AP
devSupportUsb	int	snapshot	NULL	Indicator for USB support on the AP . It could be 0 and 1. 0: not support, 1: support
deviceIpMode	int	snapshot	NULL	Current IP mode of the AP (IPv4 or IPv6)

TABLE 51 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ip	string	snapshot	NULL	IPv4 address of the AP
ipv6	string	snapshot	NULL	IPv6 address of the AP
ipsecIp	string	snapshot	NULL	IPsec virtual IP address of the AP (only populated when IPsec is enabled)
apConnectedIp	string	snapshot	NULL	IP address the AP uses to connect to the controller.
uptime	long	snapshot	NULL	Duration since the AP was last rebooted . The unit is second.
mountState	string	snapshot	NULL	AP mount state
currentTemperature	int	snapshot	NULL	Current temperature inside the AP
lifeMaxTemperature	int	snapshot	NULL	Highest AP temperature ever recorded
lifeMinTemperature	int	snapshot	NULL	Lowest AP temperature ever recorded
dnatInfo	string	snapshot	NULL	rks_gre tunnel gateway IP address
rksDpIp	string	snapshot	NULL	Data blade IP address and port number
rksDpIpOnly	string	snapshot	NULL	Data blade IP address
ipType	string	snapshot	NULL	IPv4 or IPv6
isIpTypeChanged	uint32	snapshot	NULL	Fake data; should be removed
managementVlan	uint32	snapshot	NULL	AP management VLAN ID
apState	string	snapshot	NULL	AP KPI status
isConnectionTotalCountFlagged	boolean	snapshot	NULL	AP KPI attribute "client total connection" flagging status
totalConnectedClient	uint32	snapshot	NULL	AP KPI attribute "client total connection" number
crashDump	uint32	snapshot	NULL	Indicator if there is crash dump is generated on AP or not
altitudeUnit	string	snapshot	NULL	GPS attribute : floor or meters
altitudeValue	uint32	snapshot	NULL	GPS attribute: floor value
poeMode	uint32	snapshot	NULL	8023af PoE power source
poeModeSetting	uint32	snapshot	NULL	8023af PoE mode
ipv6Type	string	snapshot	NULL	AP IPv6 mode (static, pope, auto)
freeMemoryPercentage	double	snapshot	NULL	The percentage of AP free memeory
freeStoragePercentage	double	snapshot	NULL	The percentage of AP free storage
poeUnderPowered	uint32	snapshot	NULL	AP power level
chainmask5G	string	snapshot	NULL	AP 5 G Radio Chainmask
chainmask24G	string	snapshot	NULL	AP 2.4 G Radio Chainmask
isIoTEnable	bool	snapshot	NULL	AP - IoT Enabled or Not
isUSBEnable	bool	snapshot	NULL	AP - USB Enabled or Not
isPoEOutEnable	bool	snapshot	NULL	AP - USB Enabled or Not
isSecEthEnable	bool	snapshot	NULL	AP - Secodnary Ethenet Enabled or Not
config_id	string	snapshot	NULL	Device config ID

TABLE 51 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
upd_config_id	string	snapshot	NULL	Device update config ID
config_status	.APStatusSystem.ConfigFwStatus	snapshot	NULL	Device config status
fw_upd_version	string	snapshot	NULL	Device firmware update version
fw_upd_status	.APStatusSystem.ConfigFwStatus	snapshot	NULL	Device update status
fw_upd_failed_reason	string	snapshot	NULL	Device firmware update failed reason
fw_request_id	string	snapshot	NULL	Device firmware request ID
config_failed_feature_name	string	snapshot	NULL	Device config failed feature name
config_failed_feature_attribute	string	snapshot	NULL	Device config failed feature attribute
config_failed_err_message	string	snapshot	NULL	Device config failed err message
config_failed_feature_attr_value	string	snapshot	NULL	Device config failed feature attr value
config_request_id	string	snapshot	NULL	Device config request ID
is_secure_boot_enable	int32	snapshot	NULL	secure boot status (0: disabled 1: enabled)
cpuPercentage	double	snapshot	NULL	The percentage of AP CPU using rate
totalMemory	uint64	snapshot	NULL	AP totoal memory size
freeMemory	uint64	snapshot	NULL	AP current free memory size
model	string	snapshot	NULL	AP model name
serialNumber	string	snapshot	NULL	The serial number in AP borad data
desc	string	snapshot	NULL	AP model display string
numRadio	int32	snapshot	NULL	Number of radio on AP
szConnCpIp	string	snapshot	NULL	SZ CP IPv4 address that is used by this AP connection.
szConnCpIpv6	string	snapshot	NULL	SZ CP IPv6 address that is used by this AP connection.
szConnDpIp	string	snapshot	NULL	SZ DP IPv4 address that is used by this AP connection for tunnel traffic.
szConnDpIpv6	string	snapshot	NULL	SZ DP IPv6 address that is used by thsi AP connection for tunnel traffic.
netmask	string	snapshot	NULL	The netmask is used by this AP network.
IpDnsSvr1	string	snapshot	NULL	DNS server 1 IPv4 address that is used by this AP.
IpDnsSvr2	string	snapshot	NULL	DNS server 2 IPv4 address that is used by this AP
Ipv6DnsSvr1	string	snapshot	NULL	DNS server 1 IPv6 address that is used by this AP.
Ipv6DnsSvr2	string	snapshot	NULL	DNS server 2 IPv6 address that is usd by this AP.
ApStatus	int32	snapshot	NULL	The status for AP connect to SZ
lastConfSyncTime	uint64	snapshot	NULL	The timestamp for last configuration sync up.
freeStorage	uint64	snapshot	NULL	AP free storage size

TABLE 51 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
ethPortStatus	int32	snapshot	NULL	AP ethernet port status (up/down)
rxErrorPkts	uint64	snapshot	NULL	RX error packet count on radio
txErrorPkts	uint64	snapshot	NULL	TX error packet count on radio
RxDropPkts	uint64	snapshot	NULL	RX packet drop count on radio
LanStatsRxBytes	uint64	snapshot	NULL	AP ethernet port RX data bytes
LanStatsTxBytes	uint64	snapshot	NULL	AP ethernet port TX data bytes
LanStatsRxPkts	uint64	snapshot	NULL	AP ethernet port RX data packets
LanStatsTxPkts	uint64	snapshot	NULL	AP ethernet port TX data packets
LanStatsRxErrorPkts	uint64	snapshot	NULL	AP ethernet port RX error packet count
LanStatsTxErrorPkts	uint64	snapshot	NULL	AP ethernet port TX error packet count
LanStatsRxDroppedPkts	uint64	snapshot	NULL	AP ethernet port RX drop packet count
LanStatsTxDroppedPkts	uint64	snapshot	NULL	AP ethernet port TX drop packet count
TxDropPkts	uint64	snapshot	NULL	AP total TX drop packet count on wifi

TABLE 52 LAN Port Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
port	uint32	snapshot	NULL	Port number for Ethernet interface
interface	string	snapshot	NULL	Interface name for lan device
dot1x	string	snapshot	NULL	Dot1X support mode (auth, sup, none)
logicLink	string	snapshot	NULL	Link status (up/down)
phyLink	string	snapshot	NULL	Link attributes (up/down, speed, duplex)
sfpInfo	string	snapshot	NULL	Sfp supported information
wanConnectivity	string	snapshot	NULL	Description for this interface is WAN or LAN interface
phyCapability	string	snapshot	NULL	Description for this interface capability

TABLE 53 Cable Modem Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
cmMac	string	snapshot	NULL	Cable modem device mac address
cmIp	string	snapshot	NULL	Cable modem IP address
cmFwVersion	string	snapshot	NULL	Cable modem version
cmUptime	uint32	snapshot	NULL	Cable modem alive time

TABLE 53 Cable Modem Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
cmSerialNumber	string	snapshot	NULL	Cable modem serial number
cmIpv6	string	snapshot	NULL	Cable modem IPv6 address
cmCapabilities	string	snapshot	NULL	Capabilities of cable modem
cmRangingTimeout	uint32	snapshot	NULL	Gets the data for ranging timeout
cmStatusValue	uint32	snapshot	NULL	Gets the data for status value.
cmStatusCode	string	snapshot	NULL	Gets the data for status code.
cmStatusTxPower	string	snapshot	NULL	Gets the data for *TxPower*.
cmStatusResets	uint32	snapshot	NULL	Gets the data for status reset.
cmStatusLostSyncs	uint32	snapshot	NULL	Gets the data for *StatusLostsSyncs*.
cmStatusInvalidMaps	uint32	snapshot	NULL	Gets the data *StatusInvalidMap*.
cmStatusInvalidUcds	uint32	snapshot	NULL	Gets the data *StatusInvalidUcds*.
cmStatusInvalidRangingResponses	uint32	snapshot	NULL	Gets the data *StatusInvalidRangingResponses*.
cmStatusInvalidRegistrationResponses	uint32	snapshot	NULL	Gets the data for *StatusInvalidRegistrationResponses*
cmStatusT1Timeouts	uint32	snapshot	NULL	Gets the data for *StatusT1Timeouts*.
cmStatusT2Timeouts	uint32	snapshot	NULL	Gets the data *StatusT2Timeouts*.
cmStatusT3Timeouts	uint32	snapshot	NULL	Gets the data for *StatusT3Timeouts*.
cmStatusT4Timeouts	uint32	snapshot	NULL	Gets the data for *StatusT4Timeouts*.
cmStatusRangingAborteds	uint32	snapshot	NULL	Gets the data for *statusRangingAborteds*
cmStatusDocsisOperMode	uint32	snapshot	NULL	Gets the data for *StatusDocsisOperMode*.
cmStatusModulationType	uint32	snapshot	NULL	Gets the data for *CmStatusModulationType*.
cmStatusEqualizationData	string	snapshot	NULL	Gets the data for *CmStatusEqualizationData*.

TABLE 54 AP Status LBS Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
isLBSEnable	boolean	snapshot	NULL	AP LBS supported status
isLBSConnected	boolean	snapshot	NULL	LBS connected status
isSupportLBS	boolean	snapshot	NULL	Fake data; should be removed

TABLE 55 Cellular Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
cellularWanInterface	string	snapshot	NULL	Cellular WAN interface
cellularConnectionStatus	string	snapshot	NULL	Cellular connection status
cellularIMSISIM0	string	snapshot	NULL	SIM0 IMSI

TABLE 55 Cellular Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
cellularIMSISIM1	string	snapshot	NULL	SIM1 IMSI
cellularICCIDSIM0	string	snapshot	NULL	SIM0 ICCID
cellularICCIDSIM1	string	snapshot	NULL	SIM1 ICCID
cellularIsSIM0Present	string	snapshot	NULL	SIM0 present
cellularIsSIM1Present	string	snapshot	NULL	SIM1 present
cellularTxBytesSIM0	uint64	snapshot	NULL	SIM0 Tx in Byte
cellularTxBytesSIM1	uint64	snapshot	NULL	SIM1 Tx in Byte
cellularRxBytesSIM0	uint64	snapshot	NULL	SIM0 Rx in Byte
cellularRxBytesSIM1	uint64	snapshot	NULL	SIM1 Rx in Byte
cellularActiveSim	string	snapshot	NULL	Cellular active SIM
cellularIPaddress	string	snapshot	NULL	Cellular IP address
cellularSubnetMask	string	snapshot	NULL	Cellular subnet mask
cellularDefaultGateway	string	snapshot	NULL	Cellular default gateway
cellularOperator	string	snapshot	NULL	Cellular operator
cellular3G4GChannel	int32	snapshot	NULL	Cellular 3g/4g channel
cellularSignalStrength	string	snapshot	NULL	Cellular signal strength
cellularCountry	string	snapshot	NULL	Country name
cellularRadioUptime	int32	snapshot	NULL	Cellular radio uptime
cellularLTEFirmware	string	snapshot	NULL	Cellular LTE firmware
cellularSwitchCountSIM0	int64	snapshot	NULL	SIM0 switch count
cellularSwitchCountSIM1	int64	snapshot	NULL	SIM1 switch count
cellularNWLostCountSIM0	int64	snapshot	NULL	SIM0 NW lost count
cellularNWLostCountSIM1	int64	snapshot	NULL	SIM1 NW lost count
cellularCardRemovalCountSIM0	int64	snapshot	NULL	SIM0 card removal count
cellularCardRemovalCountSIM1	int64	snapshot	NULL	SIM1 card removal count
cellularDHCPTimeoutCountSIM0	int64	snapshot	NULL	SIM0 DCHP timeout count
cellularDHCPTimeoutCountSIM1	int64	snapshot	NULL	SIM1 DCHP timeout count
cellularRoamingStatus	string	snapshot	NULL	Cellular roaming status
cellularIMEI	string	snapshot	NULL	Cellular IMEI
cellularRSRP	INT32	snapshot	NULL	Cellular RSRP
cellularRSRQ	INT32	snapshot	NULL	Cellular RSRQ
cellularSINR	INT32	snapshot	NULL	Cellular SINR
cellularRSCP	INT32	snapshot	NULL	Cellular RSCP
cellularECIO	INT32	snapshot	NULL	Cellular ECIO
cellularBand	STRING	snapshot	NULL	Cellular band
cellularUplinkBandwidth	STRING	snapshot	NULL	Cellular uplink bandwidth
cellularDownlinkBandwidth	STRING	snapshot	NULL	Cellular downlink bandwidth
gpsHistory	.GpsHistoryData	snapshot	NULL	GPS history

TABLE 56 Gps History Data

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
timestamp	UINT64	snapshot	NULL	timestamp
latitude	STRING	snapshot	NULL	latitude
longitude	STRING	snapshot	NULL	longitude

TABLE 57 AP Status Data Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
APSystem	.APStatusSystem	Serialization	NULL	Serialization data for all of AP system information.
APIPSec	.APStatusIPSec	Serialization	NULL	Serialization data for all of IPSec tunnel information.
APBrownout	.APStatusBrownout	Serialization	NULL	Serialization data for all of AP brown out information.
APRadio	.APStatusRadio	Serialization	NULL	Serialization data for all of AP radio information.
APIPSecStats	.APStatusIPSecStats	Serialization	NULL	Serialization data for all of AP IPSec tunnel stats information.
APTunnel	.APStatusTunnel	Serialization	NULL	Serialization data for all of AP tunnel stats information.
lanPortStatus	.LanPortStatus	Serialization	NULL	Serialization data for all of ethernet port status information.
cableModemInfo	.CableModemInfo	Serialization	NULL	Serialization data for all of cable modem device information.
APStatusLBS	.APStatusLBS	Serialization	NULL	Serialization data for all of LBS information.
cellularInfo	.CellularInfo	Serialization	NULL	Serialization data for all of cellular information.

TABLE 58 AP Status Data

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
version	uint32	snapshot	NULL	This stats report version number
ap_status_data	.APStatusData	Serialization	NULL	Serialization data for AP status information
zone_id	string	snapshot	NULL	Zone UUID
apgroup_id	string	snapshot	NULL	AP group UUID
cluster_id	string	snapshot	NULL	Cluster UUID
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	MAP UUID
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name

TABLE 58 AP Status Data (continued)

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
apgroup_name	string	snapshot	NULL	AP group name
domain_name	string	snapshot	NULL	Domain name
wlangroup24G_id	string	snapshot	NULL	UUID for Wlan group on 2.4G radio
wlangroup24G_name	string	snapshot	NULL	Name for wlan group on 2.4G radio
wlangroup5G_id	string	snapshot	NULL	UUID for wlan group on 5G radio
wlangroup5G_name	string	snapshot	NULL	Name for wlan group on 5G radio
sampleTime	uint64	snapshot	NULL	Timestamp to generate this stats report
aggregationInterval	uint32	snapshot	NULL	Interval for stats data aggregration
map_name	string	snapshot	NULL	MAP name
apMac	string	snapshot	NULL	AP MAC address
lacp_state	int32	snapshot	NULL	LACP Feature State 1:Enabled, 0:Disabled
wlan_groups	ApStatusWlanGroup	snapshot	NULL	General format for wlan group attributes
apRadioDeploy	string	snapshot	NULL	Indicate AP all radio band deploy

ap_wired_client.proto

```
/**
*    Copyright 2016 Ruckus Wireless, Inc. All rights reserved.
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view,use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf";

message APWiredClientInfo {
    optional string clientMac = 1;
    optional string ipAddress = 2;
    optional string ipv6Address = 3;
    optional int32 vlan = 4;
    optional uint64 rxFrames = 5;
    optional uint64 rxBytes = 6;
    optional uint64 rxUcast = 7;
    optional uint64 rxMcast = 8;
    optional uint64 rxBcast = 9;
    optional uint64 rxDrop = 10;
    optional uint64 rxEapol = 11;
    optional uint64 rxMcastLegacy = 12;
    optional uint64 txFrames = 13;
    optional uint64 txBytes = 14;
    optional uint64 txUcast = 15;
    optional uint64 txMcast = 16;
    optional uint64 txBcast = 17;
    optional uint64 txDrop = 18;
    optional uint64 txEapol = 19;
    enum AUTH_STATUS
    {
       UNAUTH = 0;
       AUTHENTICATED = 1;
    }
    optional AUTH_STATUS authStatus = 20;
    optional string ethIF = 21;
    optional string hostname = 22;
    optional int32 deviceType = 23;
    optional int32 osVendorType = 24;
    optional string modelName = 25;
    optional uint64 rxBytes_r = 26;
    optional uint64 txBytes_r = 27;
    optional int32 vni = 28;

}

message APWiredClientStats {
    optional uint32 version = 1;
    repeated APWiredClientInfo clients = 2;
    optional uint64 timestamp = 3;
    optional uint64 sampleTime = 4;
    optional uint32 aggregationInterval = 5;
    optional string zone_id = 6;
    optional string domain_id = 7;
    optional string deviceName = 8;
    optional string apgroup_id = 9;
    optional string aptenant_id = 10;
    optional string map_id = 11;
    optional string cluster_id = 12;
    optional string apmac = 13;
}
```

Field Description

TABLE 59 AP Wired Client Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
clientMac	string	Snapshot	NULL	Wired Client MAC address
ipAddress	string	Snapshot	NULL	Wired Client IPv4 address
ipv6Address	string	Snapshot	NULL	Wired Client IPv6 address
vlan	int32	Snapshot	NULL	Wired Client VLAN
rxFrames	uint64	Snapshot	NULL	Receive packet frames
rxBytes	uint64	Snapshot	NULL	Receive packet bytes
rxUcast	uint64	Snapshot	NULL	Receive unicast packets
rxMcast	uint64	Snapshot	NULL	Receive multicast packets
rxBcast	uint64	Snapshot	NULL	Receive Broadcast packets
rxDrop	uint64	Snapshot	NULL	Drop packets on receive side
rxEapol	uint64	Snapshot	NULL	Receive EAPOL packets
rxMcastLegacy	uint64	Snapshot	NULL	Receive legacy multicast packets
txFrames	uint64	Snapshot	NULL	Transmit packet frames
txBytes	uint64	Snapshot	NULL	Transmit packet bytes
txUcast	uint64	Snapshot	NULL	Transmit unicast packets
txMcast	uint64	Snapshot	NULL	Transmit multicast packets
txBcast	uint64	Snapshot	NULL	Transmit Broadcast packets
txEapol	uint64	Snapshot	NULL	Transmit EAPOL packets
authStatus	.APWiredClientInfo.A UTH_STATUS	Snapshot	NULL	Wired client authentication status(UNAUTH or AUTHENTICATED)
ethIF	string	Snapshot	NULL	Interface name which wired client associated
hostname	string	Snapshot	NULL	Client hostname
deviceType	int32	Snapshot	NULL	Client device type, such as Laptop, Smartphones, Tablets etc.
osVendorType	int32	Snapshot	NULL	Client OSVendor type, such as Windows, Android, Apple etc.
modelName	string	Snapshot	NULL	Client's device model name, such as Windows 7, Mac OS X etc.
rxBytes_r	unit64	Delta	Sum	Total bytes received
txBytes_r	unit64	Delta	Sum	Total bytes sent
vni	INT32	snapshot	NULL	Wired Client VNI

TABLE 60 AP Wired Client Stats Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	uint32	Snapshot	NULL	GPB version
clients	.APWiredClientInfo	Serialization	NULL	Serialization data for all of wired client information

TABLE 60 AP Wired Client Stats Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
timestamp	uint64	Snapshot	NULL	time for this report
sampleTime	uint64	Snapshot	NULL	sample time for these stats
aggregationInterval	uint32	Snapshot	NULL	aggregation interval for report
zone_id	string	Snapshot	NULL	zone UUID
domain_id	string	Snapshot	NULL	domain UUID
deviceName	string	Snapshot	NULL	AP device name
apgroup_id	string	Snapshot	NULL	ap group UUID
aptenant_id	string	Snapshot	NULL	ap tenant UUID
map_id	string	Snapshot	NULL	map UUID
cluster_id	string	Snapshot	NULL	cluster UUID
apmac	string	Snapshot	NULL	MAC address of the AP

commons.proto

```
syntax = "proto2";
package com.ruckuswireless.scg.protobuf.storage;

/*
 * Common used message across all users.
 */

// for common used messages
message IdList {
    repeated string id = 1;
}

// for query related messages
message ListModel {
    required int64 index = 1;
    required int64 size = 2;
    extensions 1000 to max;
}

message QueryCriteria {
    repeated Filter filters = 1;
    repeated Filter extraFilters = 2;
    repeated Filter extraNotFilters = 3;
    repeated Filter matchFilters = 4;
    optional TimeRange extraTimeRange = 5;
    optional FullTextSearch fullTextSearch = 6;
    optional Sorter sorter = 7;
    optional Page page = 8;
    optional QueryScopeInfo queryScopeInfo = 9;
}

message Filter {
    required string type = 1;
    required string value = 2;
}

message TimeRange {
    optional int64 start = 1;
    optional int64 end = 2;
    optional int64 interval = 3;
    required string field = 4;
}

message FullTextSearch {
    repeated string fields = 1;
    required SearchType searchType = 2;
    required string text = 3;
    optional MatchType matchType = 4;
}

enum MatchType {
    BEST_FIELDS = 1;
    MOST_FIELDS = 2;
    CROSS_FIELDS = 3;
    PHRASE = 4;
    PHRASE_PREFIX = 5;
}

enum SearchType {
    AND = 1;
    OR = 2;
}

message Sorter {
    required Order order = 1;
    required string field = 2;
}
```

```
enum Order {
    ASC = 1;
    DESC = 2;
}

message Page {
    required int64 index = 1;
    required int64 size = 2;
}

// RBAC related
message QueryScopeInfo {
    optional string targetResourceType = 1;
    repeated ScopeAndPermissions scopeAndPermissions = 2;
    repeated PermittedScopeAndResourceIds permittedScopeAndResourceIds = 3;
}

message PermittedScopeAndResourceIds {
    required string groupType = 1;
    repeated string resourceIds = 2;
}

message ScopeAndPermissions {
    required ResourceScope resourceScope = 1;
    repeated CategoryPermissions categoryPermissions = 2;
}

message ResourceScope {
    optional string tenantId = 1;
    optional string domainId = 2;
    optional string zoneId = 3;
}

// resourceType: "WLAN_CATEGORY", operationType: "FULL_ACCESS"
message CategoryPermissions {
    required string resourceType = 1;
    required string operationType = 2;
}

// Aggregation
enum AggregationType {
    AVG = 1;
    CARDINALITY = 2;
    COUNT = 3;
    HISTOGRAM = 4;
    MAX = 5;
    MIN = 6;
    SUM = 7;
    TERM = 8;
    DATE_HISTOGRAM = 9;
}

message AggregationRequest {
    required QueryCriteria queryCriteria = 1;
    required Aggregation aggregation = 2;
}

message Aggregation {
    required string field = 1;
    required AggregationType type = 2;
    // for date histogram aggregation
    optional int64 startTimestamp = 3;
    optional int64 endTimestamp = 4;
    optional int64 interval = 5;
    // for terms aggregation
    optional int32 size = 6;
    repeated SubAggregation subAggregations = 7;
}

message SubAggregation {
    required string field = 1;
    required AggregationType type = 2;
```

```
}

message AggregationResult {
    repeated AggregationBucket buckets = 1;
}

message AggregationBucket {
    required string name = 1;
    required bool hasValue = 2;
    optional string literalValue = 3;
    optional double numericValue = 4;
    repeated AggregationBucket subBuckets = 5;
}
```

nanopb.proto

```
// Custom options for defining:
// - Maximum size of string/bytes
// - Maximum number of elements in array
//
// These are used by nanopb to generate statically allocable structures
// for memory-limited environments.

syntax = "proto2";
import "google/protobuf/descriptor.proto";

option java_package = "fi.kapsi.koti.jpa.nanopb";

enum FieldType {
    FT_DEFAULT = 0; // Automatically decide field type, generate static field if possible.
    FT_CALLBACK = 1; // Always generate a callback field.
    FT_POINTER = 4; // Always generate a dynamically allocated field.
    FT_STATIC = 2; // Generate a static field or raise an exception if not possible.
    FT_IGNORE = 3; // Ignore the field completely.
    FT_INLINE = 5; // Legacy option, use the separate 'fixed_length' option instead
}

enum IntSize {
    IS_DEFAULT = 0; // Default, 32/64bit based on type in .proto
    IS_8 = 8;
    IS_16 = 16;
    IS_32 = 32;
    IS_64 = 64;
}

// This is the inner options message, which basically defines options for
// a field. When it is used in message or file scope, it applies to all
// fields.
message NanoPBOptions {
  // Allocated size for 'bytes' and 'string' fields.
  // For string fields, this should include the space for null terminator.
  optional int32 max_size = 1;

  // Maximum length for 'string' fields. Setting this is equivalent
  // to setting max_size to a value of length+1.
  optional int32 max_length = 14;

  // Allocated number of entries in arrays ('repeated' fields)
  optional int32 max_count = 2;

  // Size of integer fields. Can save some memory if you don't need
  // full 32 bits for the value.
  optional IntSize int_size = 7 [default = IS_DEFAULT];

  // Force type of field (callback or static allocation)
  optional FieldType type = 3 [default = FT_DEFAULT];

  // Use long names for enums, i.e. EnumName_EnumValue.
  optional bool long_names = 4 [default = true];

  // Add 'packed' attribute to generated structs.
  // Note: this cannot be used on CPUs that break on unaligned
  // accesses to variables.
  optional bool packed_struct = 5 [default = false];

  // Add 'packed' attribute to generated enums.
  optional bool packed_enum = 10 [default = false];

  // Skip this message
  optional bool skip_message = 6 [default = false];

  // Generate oneof fields as normal optional fields instead of union.
  optional bool no_unions = 8 [default = false];
```

```
  // integer type tag for a message
  optional uint32 msgid = 9;

  // decode oneof as anonymous union
  optional bool anonymous_oneof = 11 [default = false];

  // Proto3 singular field does not generate a "has_" flag
  optional bool proto3 = 12 [default = false];

  // Generate an enum->string mapping function (can take up lots of space).
  optional bool enum_to_string = 13 [default = false];

  // Generate bytes arrays with fixed length
  optional bool fixed_length = 15 [default = false];
}

// Extensions to protoc 'Descriptor' type in order to define options
// inside a .proto file.
//
// Protocol Buffers extension number registry
// --------------------------------
// Project:  Nanopb
// Contact:  Petteri Aimonen <jpa@kapsi.fi>
// Web site: http://kapsi.fi/~jpa/nanopb
// Extensions: 1010 (all types)
// --------------------------------

extend google.protobuf.FileOptions {
    optional NanoPBOptions nanopb_fileopt = 1010;
}

extend google.protobuf.MessageOptions {
    optional NanoPBOptions nanopb_msgopt = 1010;
}

extend google.protobuf.EnumOptions {
    optional NanoPBOptions nanopb_enumopt = 1010;
}

extend google.protobuf.FieldOptions {
    optional NanoPBOptions nanopb = 1010;
}
```

ScgSessMgrPubIpc.proto

```
package ScgSessMgrPubIpc;

option java_package = "com.ruckuswireless.scg.protobuf.sessmgr";
option java_outer_classname = "ScgSessMgrPubIpc";

message GBP_SESSMGR_PUBS_STATS_UE_PMIP_SESS_STAT
{
    optional string   relay_mac = 1;
    optional string   mn_mac = 2;
    optional string   hoa_addr = 3;
    optional string   magControlIp = 4; //Not a manditory parameter.
    optional string   ap_mac = 5;
    optional string   relay_netaddr = 6;
    optional string   lma_addr = 7;
    optional string   mvno_id = 8;
    optional string   cause = 9;
    optional uint64   start_time = 10;
};

message GBP_SESSMGR_OBS_LMA_SIG_STATS
{
       optional string dbladeId = 1;
       optional string lmaIp    = 2;
       optional string mvnoId   = 3;
       optional uint32 pbuPkts  = 4;
       optional uint32 pbaPkts  = 5;
       optional uint32 pbuLifetime0Pkts = 6;
       optional uint32 pbaLifetime0Pkts = 7;
       optional uint32 briPkts  = 8;
       optional uint32 braPkts  = 9;
       optional uint32 totalPkts = 10;
       optional uint64 recordCreationTime = 11;
       optional uint64 recordUpdateTime   = 12;
};

message GBP_SESSMGR_OBS_LMA_CON_STATS
{
      optional string dbladeId = 1;
      optional string mvnoId   = 2;
      optional string primaryLMAIp = 3;
      optional string secondaryLMAIp = 4;
      optional string activeLMAIp   = 5;
      optional uint32 primaryLMADuration = 6;
      optional uint32 secondaryLMADutation = 7;
      optional uint32 numOfFailOverPrimaryToSecondary = 8;
      optional uint32 numOfFailOverSecondaryToPrimary = 9;
      optional uint32 lastFailOverTime = 10;
      optional uint32 recordCreationTime = 11;
      optional uint32 recordUpdateTime  = 12;
};

message GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T
{
   optional string apMac = 1;
   optional string apIP = 2;
   optional string clientMac = 3;
   optional string ipAddress = 4;
   optional string apName = 5;
   optional string rssi = 6;
   optional string signal = 7;
   optional string txRSSI = 8;
   optional string mobilityZone = 9;
   optional string wlanId = 10;
   optional string vlan = 11;
   optional string mode = 12;
   optional string ssid = 13;
   optional string channel = 14;
   optional string radio = 15;
```

```
    optional uint32 status = 16;
    optional string zoneId = 17;
    optional string thirdPWlanID = 18;
    optional string encryptMethod = 19;
    optional string authmethod = 20;
    optional string user = 21;
    optional string bssid = 22;
    optional string fwdPolicy = 23;
    optional uint64 connectSince = 24;
    optional uint32 txRetry = 25;
    optional uint64 rxFrames = 26;
    optional uint64 txFrames = 27;
    optional uint64 rxBytes = 28;
    optional uint64 txBytes = 29;
    optional uint32 location = 30;
    optional uint32 termCause = 31;
    optional uint32 ttlVal = 32;
    optional string hostname = 33;
    optional string dvcinfo = 34;
    optional bytes  dvcType = 35;
    optional string ifname = 36;
    optional uint32 sessStartTime = 37;
    optional uint64 rxDrops = 38;
    optional uint64 txDrops = 39;
    optional string mvnoId  = 40;
    optional string ueIpAddr_ipv6 = 41;
    optional uint32 sessEndTime = 42;
    optional string domainId = 43;
    enum UE_TYPE
     {
        WIRELESS = 0;
        WIRED    = 1;
     }
    optional UE_TYPE ueType = 44 [default = WIRELESS];
    optional uint32 ethProfId = 45;
    optional bool isThirdParty = 46;
    optional bool isTTG = 47;
    optional string model_name = 48;
    optional bytes  device_type_id = 49;
    optional bytes  os_vendor_id = 50;
    optional string acctSessId = 51;
    optional string acctMultiSessId = 52;
    optional uint32 mlisaState = 53;
    optional string userRoleId = 54;
};

message GBP_SESSMGR_PUBS_STATS_UE_PER_SESS_STATS
{
    optional string ueMac = 1;
    optional uint64 ueIpAddr = 2;
    optional string ueImsi = 3;
    optional string ueCui = 4;
    optional uint64 ueSessStartTime = 5;
    optional uint64 ueSessEndTime = 6;
    optional uint64 ueSessType = 7;
    optional uint64 ueSessState = 8;
    optional uint64 ueChargType = 9;
    optional uint64 chargSerIp = 10;
    optional string termCause = 11;
    optional uint64 authSerIp = 12;
    optional uint64 authType = 13;
    optional string ueRealm = 14;
    optional string cBladeId = 15;
    optional uint64 ueMnc = 16;
    optional uint64 ueMcc = 17;
    optional string mvnoId = 18;
    optional uint64 wlanId = 19;
    optional string zoneId = 20;
    optional string thirdPartyApZoneId = 21;
    optional string ssId = 22;
    optional string apMac = 23;
    optional uint64 chrgSrvrType = 24;
```

```
    optional string chrgSrvcName = 25;
    optional uint64 authSrvrType = 26;
    optional string  authSrvcName = 27;
    optional uint32 ttlVal = 28;
    optional string ueIpAddr_ipv6 = 29;
}

message GBP_SESSMGR_PUBS_STATS_TTG_INFO_SESS_STATS
{
    optional string ueMac = 1;
    optional uint64 ueIpAddr = 2;
    optional uint64 ueSessStartTime = 3;
    optional string ueApn = 4;
    optional uint64 ggsnIp = 5;
    optional uint64 uePdpIp = 6;
    optional string cBladeId = 7;
    optional uint64 recCreateTime = 8;
    optional uint32 ttlVal = 9;
    optional string ueIpAddr_ipv6 = 10;
}

message GBP_SESSMGR_PUBS_STATS_PDP_CTXT_INFO_SESS_STATS
{
    optional string ueMac = 1;
    optional uint64 ueIpAddr = 2;
    optional uint64 ueSessStartTime = 3;
    optional uint64 gtpCIp = 4;
    optional uint64 gtpDIp = 5;
    optional uint64 gtpCUpTeid = 6;
    optional uint64 gtpCDownTeid = 7;
    optional uint64 gtpUUpTeid = 8;
    optional uint64 gtpUDownTeid = 9;
    optional string cBladeId = 10;
    optional string gtpQos = 11;
    optional uint64 recCreateTime = 12;
    optional uint64 pdpStatus = 13;
    optional uint64 ggsnUIp = 14;
    optional uint32 ttlVal = 15;
    optional string ueIpAddr_ipv6 = 16;
}

message GBP_SESSMGR_PUBS_STATS_THIRDPAR_AP_STATS
{
    optional string cBladeId  = 1;
    optional string mvnoId = 2;
    optional string apMac  = 3;
    optional string  apZone = 4;
    optional uint64 apIpAddr = 5;
    optional uint64 numClnts = 6;
    optional uint64 lastSeenTime = 7;
    optional uint64 recCrtTime = 8;
    optional uint64 recUpdTime = 9;
}

message GBP_SESSMGR_PUBS_STATS_THIRDPAR_CLIENT_STATS
{
    optional string cBladeId = 1;
    optional string mvnoId = 2;
    optional string apMac = 3;
    optional uint64 apIpAddr = 4;
    optional string clientMac = 5;
    optional uint64 clientIpAddr = 6;
    optional string ssid = 7;
    optional uint64 authMethod = 8;
    optional string  apZone = 9;
    optional string  userName = 10;
    optional uint64 recCrtTime = 11;
    optional uint64 recUpdTime = 12;
    optional uint32 ttlVal = 13;
    optional bytes  sessStatus = 14;
}
```

```
enum GBP_SESSMGR_DPSK_OP_TYPE
{
   GBP_SESSMGR_DPSK_OP_TYPE_ADD = 1;
   GBP_SESSMGR_DPSK_OP_TYPE_MOD = 2;
   GBP_SESSMGR_DPSK_OP_TYPE_DEL = 3;
   GBP_SESSMGR_DPSK_OP_TYPE_DEL_SHM = 4;
}

enum GBP_SESSMGR_PUBS_DPSK_EXPIRATION_START_POINT
{
    FROM_CREATION = 1;
    FROM_FIRST_USE = 2;
}

message GBP_SESSMGR_PUBS_DPSK_ENTRY
{
   optional GBP_SESSMGR_DPSK_OP_TYPE opType = 1;
   optional string key = 2; // uuid
   optional uint64 ueMac = 3;
   optional string dpsk = 4;
   optional string prevDpsk = 5;
   optional uint32 bounded = 6;
   optional uint32 vlanId = 7;
   optional uint32 wlanId = 8;
   optional string zoneId = 9;
   optional string tenantId = 10;
   optional string userRoleId = 11;
   optional string utpId = 12;
   optional string userName = 13;
   optional GBP_SESSMGR_PUBS_DPSK_EXPIRATION_START_POINT expirationStartPoint = 14;
   optional uint64 expirationStartTime = 15; // Creation time or first use time of this DPSK entry.
epoch time in millisec. When FROM_FIRST_USE, filled in by SessManager.
   optional uint64 ttl = 16; // Unit: Minutes; 0 means Never Expired.
   optional uint32 maxDevice = 17; // For unbound case that one DPSK can be shared by multiple device.
   optional uint32 deviceCounting = 18; // For unbound case; indicate how many device is using this DPSK
   optional string pmk = 19;
}

message GBP_SESSMGR_PUBS_MULTIPLE_DPSK_ENTRIES
{
   repeated GBP_SESSMGR_PUBS_DPSK_ENTRY dpsks = 1;
}

enum GBP_IDM_DPSK_OP_TYPE
{
   GBP_IDM_DPSK_OP_TYPE_ADD = 1;
   GBP_IDM_DPSK_OP_TYPE_MOD = 2;
   GBP_IDM_DPSK_OP_TYPE_DEL = 3;
}

message GBP_IDM_PUBS_DPSK_ENTRY
{
   optional GBP_IDM_DPSK_OP_TYPE opType = 1;
   optional string key = 2; // uuid
   optional uint64 ueMac = 3;
   optional string dpsk = 4;
   optional string prevDpsk = 5;
   optional uint32 vlanId = 6;
   optional uint32 wlanId = 7;
   optional string zoneId = 8;
   optional string tenantId = 9;
   optional string userRoleId = 10;
   optional uint32 utpId = 11;  // 0~63 UTP ID
   optional string userName = 12;

   enum GBP_IDM_DPSK_EXPIRATION_START_POINT
   {
       FROM_CREATION = 1;
       FROM_FIRST_USE = 2;
   }
   optional GBP_IDM_DPSK_EXPIRATION_START_POINT expirationStartPoint = 13;
   optional uint64 expirationStartTime = 14; // Creation time or first use time of this DPSK entry.
```

```
epoch time in millisec. When FROM_FIRST_USE, filled in by SessManager.
   optional uint64 ttl = 15; // Unit: Minutes; 0 means Never Expired.
   optional string pmk = 16;
   optional bool   isGroup = 17; // for Group DPSK
}

message GBP_IDM_PUBS_MULTIPLE_DPSK_ENTRIES
{
   optional GBP_IDM_DPSK_OP_TYPE opType = 1;
   repeated GBP_IDM_PUBS_DPSK_ENTRY dpsks = 2;
}

message GBP_SESSMGR_PUBS_EVT_UE_JOIN
{
    optional uint64 ueMac = 1;
    optional uint64 apMac = 2;
    optional bytes  apBssId = 3;
    optional uint32 wlanId = 4;
    optional uint32 dpIP = 5;
    optional uint32 cpIP = 6;
}

message GBP_SESSMGR_PUBS_EVT_UE_ROAM
{
    optional uint64 ueMac = 1;
    optional uint64 apMac = 2;
    optional bytes  apBssId = 3;
    optional uint32 wlanId = 4;
    optional uint64 oldapMac = 5;
    optional uint32 dpIP = 6;
    optional uint32 cpIP = 7;
}

message GBP_SESSMGR_PUBS_EVT_UE_AUTHD
{
    optional uint64 ueMac = 1;
    optional uint64 apMac = 2;
    optional bytes  apBssId = 3;
    optional uint32 wlanId = 4;
    optional uint32 authType = 5;
    optional uint32 authMethod = 6;
    optional uint32 dpIP = 7;
    optional uint32 cpIP = 8;
}

message GBP_SESSMGR_PUBS_EVT_UE_UNAUTHD
{
    optional uint64 ueMac = 1;
    optional uint64 apMac = 2;
    optional bytes  apBssId = 3;
    optional uint32 wlanId = 4;
    optional uint32 authType = 5;
    optional uint32 authMethod = 6;
    optional uint32 dpIP = 7;
    optional uint32 cpIP = 8;
}

message GBP_SESSMGR_PUBS_EVT_UE_LEAVE
{
    optional uint64 ueMac = 1;
    optional uint64 apMac = 2;
    optional bytes  apBssId = 3;
    optional uint32 wlanId = 4;
    optional uint32 dpIP = 5;
    optional uint32 cpIP = 6;
    optional uint32 termCause =7;
}

message GBP_SESSMGR_PUBS_OBR_LOG_LEVEL
{
    optional uint32 logLevel = 1;
}
```

```
enum GBP_SESSMGR_PUBS_MSG_TYPE
{
    GBP_SESSMGR_PUBS_MSG_TYPE_PMIP_SESS_STATS = 1;
    GBP_SESSMGR_PUBS_MSG_TYPE_AP_CLIENTS_STATS = 2;
    GBP_SESSMGR_PUBS_MSG_TYPE_UE_PER_SESS_STATS = 3;
    GBP_SESSMGR_PUBS_MSG_TYPE_TTG_INFO_STATS = 4;
    GBP_SESSMGR_PUBS_MSG_TYPE_PDP_CTXT_STATS = 5;
    GBP_SESSMGR_PUBS_MSG_TYPE_3P_STATS = 6;
    GBP_SESSMGR_PUBS_MSG_TYPE_3P_CLIENTS_STATS = 7;
    GBP_SESSMGR_PUBS_MSG_TYPE_DPSK_ENTRY=8;
    GBP_SESSMGR_PUBS_MSG_TYPE_BULK_DPSK_ENTRY=9;
    SESSMGR_PUBS_MSG_TYPE_PMIPV6_LMA_SIGNAL_ENTRY = 10;
    SESSMGR_PUBS_MSG_TYPE_PMIPV6_LMA_CONNECTIVITY_ENTRY = 11;
    GBP_IDM_PUBS_MSG_TYPE_DPSK_ENTRY = 12;
    GBP_IDM_PUBS_MSG_TYPE_BULK_DPSK_ENTRY = 13;
    GBP_SESSMGR_PUBS_EVT_TYPE_UE_JOIN = 14;
    GBP_SESSMGR_PUBS_EVT_TYPE_UE_ROAM = 15;
    GBP_SESSMGR_PUBS_EVT_TYPE_UE_AUTHD = 16;
    GBP_SESSMGR_PUBS_EVT_TYPE_UE_UNAUTHD = 17;
    GBP_SESSMGR_PUBS_EVT_TYPE_UE_LEAVE = 18;
    GBP_SESSMGR_PUBS_MSG_TYPE_OBR_LOG_LEVEL = 19;
}

message GBP_SESSMGR_PUB_MSG
{
    required GBP_SESSMGR_PUBS_MSG_TYPE pubMsgType = 1;
    optional GBP_SESSMGR_PUBS_STATS_THIRDPAR_CLIENT_STATS pub3PApClientsStats = 2;
    optional GBP_SESSMGR_PUBS_STATS_PDP_CTXT_INFO_SESS_STATS pubPdpCtxtStats = 3;
    optional GBP_SESSMGR_PUBS_STATS_THIRDPAR_AP_STATS   pub3PApStats = 4;
    optional GBP_SESSMGR_PUBS_STATS_TTG_INFO_SESS_STATS pubTtgInfoStats = 5;
    optional GBP_SESSMGR_PUBS_STATS_UE_PER_SESS_STATS   pubUePerSessStats = 6;
    optional GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T    pubApClientsStats = 7;
    optional GBP_SESSMGR_PUBS_STATS_UE_PMIP_SESS_STAT   pubPmipSessStats = 8;
    optional GBP_SESSMGR_PUBS_DPSK_ENTRY                pubDpskEntry=9;
    optional GBP_SESSMGR_PUBS_MULTIPLE_DPSK_ENTRIES     pubDpskBulkEntry=10;
    optional GBP_SESSMGR_OBS_LMA_SIG_STATS              pubPmipLmaSigStats = 11;
    optional GBP_SESSMGR_OBS_LMA_CON_STATS              pubPmipConSigStats = 12;
    optional GBP_IDM_PUBS_DPSK_ENTRY                    pubIdmDpskEntry = 13;
    optional GBP_IDM_PUBS_MULTIPLE_DPSK_ENTRIES         pubIdmDpskBulkEntry = 14;
    optional GBP_SESSMGR_PUBS_EVT_UE_JOIN               pubevtUeJoin = 15;
    optional GBP_SESSMGR_PUBS_EVT_UE_ROAM               pubevtUeRoam = 16;
    optional GBP_SESSMGR_PUBS_EVT_UE_AUTHD              pubevtUeAuthd = 17;
    optional GBP_SESSMGR_PUBS_EVT_UE_UNAUTHD            pubevtUeUnauthd = 18;
    optional GBP_SESSMGR_PUBS_EVT_UE_LEAVE              pubevtUeLeave = 19;
    optional GBP_SESSMGR_PUBS_OBR_LOG_LEVEL             pubObrLogLevel = 20;
}
```

Field Description

TABLE 61 GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
apMac	string	snapshot	NULL	
apIP	string	snapshot	NULL	
clientMac	string	snapshot	NULL	
ipAddress	string	snapshot	NULL	
apName	string	snapshot	NULL	
rssi	string	snapshot	NULL	
signal	string	snapshot	NULL	
txRSSI	string	snapshot	NULL	

TABLE 61 GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
mobilityZone	string	snapshot	NULL	
wlanId	string	snapshot	NULL	
vlan	string	snapshot	NULL	
mode	string	snapshot	NULL	
ssid	string	snapshot	NULL	
channel	string	snapshot	NULL	
radio	string	snapshot	NULL	
status	uint32	snapshot	NULL	
zoneId	string	snapshot	NULL	
thirdPWlanID	string	snapshot	NULL	
encryptMethod	string	snapshot	NULL	
authmethod	string	snapshot	NULL	
user	string	snapshot	NULL	
bssid	string	snapshot	NULL	
fwdPolicy	string	snapshot	NULL	
connectSince	uint64	snapshot	NULL	
txRetry	uint32	snapshot	NULL	
rxFrames	uint64	snapshot	NULL	
txFrames	uint64	snapshot	NULL	
rxBytes	uint64	snapshot	NULL	
txBytes	uint64	snapshot	NULL	
location	uint32	snapshot	NULL	
termCause	uint32	snapshot	NULL	
ttlVal	uint32	snapshot	NULL	
hostname	string	snapshot	NULL	
dvcinfo	string	snapshot	NULL	
dvcType	bytes	snapshot	NULL	
ifname	string	snapshot	NULL	
sessStartTime	uint32	snapshot	NULL	
rxDrops	uint64	snapshot	NULL	
txDrops	uint64	snapshot	NULL	
mvnoId	string	snapshot	NULL	
ueIpAddr_ipv6	string	snapshot	NULL	
sessEndTime	uint32	snapshot	NULL	
domainId	string	snapshot	NULL	
ueType	UE_TYPE	snapshot	NULL	
ethProfId	uint32	snapshot	NULL	
isThirdParty	bool	snapshot	NULL	
isTTG	bool	snapshot	NULL	
model_name	string	snapshot	NULL	

TABLE 61 GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
device_type_id	bytes	snapshot	NULL	
os_vendor_id	bytes	snapshot	NULL	
acctSessId	string	snapshot	NULL	
acctMultiSessId	string	snapshot	NULL	
mlisaState	uint32	snapshot	NULL	
userRoleId	string	userRoleId	NULL	User role of the client belongs to.

sci-alarm.proto

```
*Copyright 2013 Ruckus Wireless, Inc. All rights reserved.
*
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view, use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*
*    JsonMessage GPB format is used to transfer the JSON messages across
*    applications which includes the version, message content and message
*    content type attributes.
*/
option java_package = "com.ruckuswireless.scg.protobuf.sci";

message AlarmMessage {
    enum AlarmState {
             OUTSTANDING = 0;
             CLEARED = 1;
    }
    optional uint32 version = 1;
    optional string alarmUuid = 2;
    optional uint32 alarmCode = 3;
    optional string alarmSeverity = 4;
    optional string mainCategory = 5;
    optional string alarmType = 6;
    optional uint32 initEventCode = 7;
    optional uint64 timestamp = 8;
    optional AlarmState alarmState = 9;
    repeated AlarmMessageEntry attributes = 10;
    optional string domainId = 11;
    optional string zoneId = 12;
    optional string apGroupId = 13;
    optional string apMac = 14;
    optional string clientMac = 15;
    optional string reason = 16;
    optional string domainName = 17;
    optional string zoneName = 18;
    optional string apGroupName = 19;
    optional string apIpAddress = 20;
    optional string apIpv6Address = 21;
    optional string description = 22;
    optional string subCategory = 23;
    optional string bladeId = 24;
}

message AlarmMessageEntry {
    optional string key = 1;
    optional string value = 2;
}
```

Field Description

TABLE 62 Enum Alarm Message Alarm State

Name	Value	Description
OUTSTANDING	0	OUTSTANDING
CLEARED	1	CLEARED

TABLE 63 Alarm Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
alarmUuid	string	snapshot	NULL	Alarm UUID
alarmCode	uint32	snapshot	NULL	Alarm Code defined by AlarmEnum
alarmSeverity	string	snapshot	NULL	Alarm severity defined by AlarmEnum
mainCategory	string	snapshot	NULL	Alarm main categorty defined by AlarmCategory
alarmType	string	snapshot	NULL	Alarm type description defined by AlarmEnum represented with String.
initEventCode	uint32	snapshot	NULL	The triggerring event code number for this alarm.
timestamp	uint64	snapshot	NULL	The timestamp of this alarm's creation.
alarmState	.AlarmMessage.AlarmState	snapshot	NULL	The AlarmState of current triggering alarm which contains (OUTSTANDING/CLEARED).
attributes	.AlarmMessageEntry	snapshot	NULL	The AlarmMessageEntry contains additional attribute values required by outer service.
domainId	string	snapshot	NULL	Domain UUID.
zoneId	string	snapshot	NULL	Zone UUID. Unique zone ID assigned by SZ controller.
apGroupId	string	snapshot	NULL	AP Group identifier.
apMac	string	snapshot	NULL	Access Point MAC address.
clientMac	string	snapshot	NULL	UE/Client MAC address if the events are related to client.
reason	string	snapshot	NULL	Reason for the alarm to occur.
domainName	string	snapshot	NULL	Domain name.
zoneName	string	snapshot	NULL	Zone name assigned by the controller. The admin configures the Zone name via the contoller's user interface.
apGroupName	string	snapshot	NULL	AP Group name.
apIpAddress	string	snapshot	NULL	IPv4 address of the AP.
apIpv6Address	string	snapshot	NULL	IPv6 address of the AP.
description	string	snapshot	NULL	Description of the alarm represented by string.
subCategory	string	snapshot	NULL	Sub category of the alarm.
bladeId	string	snapshot	NULL	Blade ID information from which node alarm created.

TABLE 64 Alarm Message Entry Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
key	string	snapshot	NULL	Alarm message entry key.
value	string	snapshot	NULL	Alarm message entry value.

sci configuration message

```
*Copyright 2013 Ruckus Wireless, Inc. All rights reserved.
*
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view, use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*/
option java_package = "com.ruckuswireless.scg.protobuf.sci";

message ConfigurationMessage {
    required uint32 version = 1;
    optional ClusterMessage clusterInfo = 2;
    optional uint64 timestamp = 3;
}

message ClusterMessage {
    /* Cluster Informations */
    optional string clusterUuid = 1;
    optional string clusterName = 2;
    optional string controlBlades = 3; // (/wsg/api/scg/planes/control
and /wsg/api/sci/cbs)
    optional string controllerUtilizations = 4; //
(/wsg/api/sci/cbutils)
    optional string systemSummary = 5; //
(/wsg/api/scg/planes/systemSummary)
    /* Raw Data [Compression]*/
    optional string domains = 100; //
(/wsg/api/scg/session/currentUser/domainList)
    optional string zones = 101; //
(/wsg/api/scg/zones/byDomain/$domain)
    optional string apGroups = 102; //
(/wsg/api/scg/apgroup/byZone/$zone)
    optional string wlanGroups = 103; //
(/wsg/api/scg/wlangroup/byZone/$zone)
    optional string wlans = 104; // (/wsg/api/scg/wlans/byZone/$zone)
    optional string aps = 105; // (/wsg/api/sci/aps)
    /* System Hierarchy [Group Tree]*/
    repeated TenantMessage tenantInfos = 200;
}

message TenantMessage {
    optional string tenantId = 1;
    optional string tenantName = 2;
    optional DomainMessage adminDomain = 3;
}

message DomainMessage {
    optional string domainId = 1;
    optional string domainName = 2;
    repeated ZoneMessage zoneInfos = 3;
    repeated DomainMessage subDomainInfos = 4;
}

message ZoneMessage {
    optional string zoneId = 1;
    optional string zoneName = 2;
    repeated ApGroupMessage apGroupInfos = 3;
    repeated WlanGroupMessage wlanGroupInfos = 4;
}

message ApGroupMessage {
    optional string apGroupId = 1;
    optional string apGroupName = 2;
}

message WlanGroupMessage {
    optional string wlanGroupId = 1;
```

```
    optional string wlanGroupName = 2;
}
```

Field Description

TABLE 65 Configuration Message Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
version	uint32	snapshot	NULL	GPB version
clusterInfo	.ClusterMessage	snapshot	NULL	Cluster information
timestamp	uint64	snapshot	NULL	Time and date of cluster message

TABLE 66 Cluster Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
clusterUuid	string	snapshot	NULL	Cluster UUID
clusterName	string	snapshot	NULL	Cluster name
controlBlades	string	snapshot	NULL	JSON string of control node information
controllerUtilizations	string	snapshot	NULL	JSON string of system resource utilization
systemSummary	string	snapshot	NULL	JSON string of system summary
domains	string	snapshot	NULL	JSON string of domain list
zones	string	snapshot	NULL	JSON string of zone list
apGroups	string	snapshot	NULL	JSON string of AP group list
wlans	string	snapshot	NULL	JSON string of WLAN list
aps	string	snapshot	NULL	JSON string of AP list
tenantInfos	.TenantMessage	snapshot	NULL	Group Tree System Hierarchy

TABLE 67 Tenant Message Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
tenantId	string	snapshot	NULL	Tenant UUID
tenantName	string	snapshot	NULL	Tenant name
adminDomain	.DomainMessage	snapshot	NULL	Domain information

TABLE 68 Domain Message Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
domainId	string	snapshot	NULL	Domain UUID
domainName	string	snapshot	NULL	Domain name
zoneInfos	.ZoneMessage	snapshot	NULL	Zone information

TABLE 68 Domain Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
subDomainInfos	.DomainMessage	snapshot	NULL	Sub Domain information

TABLE 69 Zone Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
zoneId	string	snapshot	NULL	Zone UUID
zoneName	string	snapshot	NULL	Zone name
apGroupInfos	.ApGroupMessage	snapshot	NULL	AP Group information
wlanGroupInfo	.WlanGroupMessage	snapshot	NULL	WLAN Group information

TABLE 70 AP Group Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
apGroupId	string	snapshot	NULL	AP Group UUID
apGroupName	string	snapshot	NULL	AP Group name

TABLE 71 WLAN Group Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
wlanGroupId	string	snapshot	NULL	WLAN Group UUID
wlanGroupName	string	snapshot	NULL	WLAN Group name

sci_event.proto

```
*Copyright 2013 Ruckus Wireless, Inc. All rights reserved.
*
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view, use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*
*    JsonMessage GPB format is used to transfer the JSON messages across
*    applications which includes the version, message content and message
*    content type attributes.
*/
option java_package = "com.ruckuswireless.scg.protobuf.sci";
message EventMessage {
    required uint32 version = 1;
    optional uint32 eventCode = 2;
    optional string eventType = 3;
    optional string mainCategory = 4;
    optional string subCategory = 5;
    optional string domainId = 6;
    optional string zoneId = 7;
    optional string apGroupId = 8;
    optional string apMac = 9;
    optional string clientMac = 10;
    optional uint64 timestamp = 11;
    repeated MessageEntry attributes = 12;
    optional string severity = 13;
    optional string reason = 14;
    optional string domainName = 15;
    optional string zoneName = 16;
    optional string apGroupName = 17;
    optional string apIpAddress = 18;
    optional string apIpv6Address = 19;
    optional string description = 20;
    optional string disconnectReason = 21;
    optional string tenantId = 22;
    optional string switchId = 23;
    optional string switchUnitId = 24;
    optional string switchGroupId = 25;
    optional string switchGroupLevelOneId = 26;
    optional string switchGroupLevelTwoId = 27;
    optional string radio = 28;
    optional string rssi = 29;
    optional string band = 30;
    optional string channel = 31;
    optional string rfband = 32;
}
message MessageEntry {
    optional string key = 1;
    optional string value = 2;
}
```

Field Description

TABLE 72 Event Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
eventCode	uint32	snapshot	NULL	Event code
eventType	string	snapshot	NULL	Event type
mainCategory	string	snapshot	NULL	Event main category

TABLE 72 Event Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
subCategory	string	snapshot	NULL	Event sub category
domainId	string	snapshot	NULL	Domain UUID
zoneId	string	snapshot	NULL	Zone UUID. Unique zone ID (for example, b381206b-2e5d-43dc-b249-e36ffae9855c) assigned by the controller.
apGroupId	string	snapshot	NULL	AP Group identifier
apMac	string	snapshot	NULL	Access Point MAC address
clientMac	string	snapshot	NULL	UE/Client MAC address if the events are related to the client
timestamp	uint64	snapshot	NULL	Event timestamp (UTC time)
attributes	.MessageEntry	snapshot	NULL	Additional event attributes.
severity	string	snapshot	NULL	Event severity
reason	string	snapshot	NULL	Reason for the event to occur.
domainName	string	snapshot	NULL	Domain name.
zoneName	string	snapshot	NULL	Zone name. Zone name assigned by the controller. The admin configures the Zone name via the controller's user interface. The controller passes it to the AP. AP retains the zone name.
apGroupName	string	snapshot	NULL	AP Group name
apIPAddress	string	snapshot	NULL	IPv4 address of the AP
apIPv6Address	string	snapshot	NULL	IPv6 address of the AP
description	string	snapshot	NULL	Description of the event
disconnectReason	string	snapshot	NULL	Disconnect reason of the event
tenantId	string	snapshot	NULL	The tenant id of switch
switchId	string	snapshot	NULL	Switch ID
switchUnitId	string	snapshot	NULL	Switch unit ID
switchGroupId	string	snapshot	NULL	Group identifier of switch.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
radio	string	snapshot	NULL	Radio
rssi	string	snapshot	NULL	Rssi
band	string	snapshot	NULL	Band
channel	string	snapshot	NULL	Channel
rfband	string	snapshot	NULL	rfband

TABLE 73 Message Entry Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
key	string	snapshot	NULL	Message key
value	string	snapshot	NULL	Message content

sci-message.proto

```
*Copyright 2017 Ruckus Wireless, Inc. All rights reserved.
*
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view, use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*
*    JsonMessage GPB format is used to transfer the JSON messages across
*    applications which includes the version, message content and message
*    content type attributes.
*/
option java_package = "com.ruckuswireless.scg.protobuf.sci";

option java_outer_classname = "SciProtocolMessage";
import "ap_status.proto";
import "ap_report.proto";
import "ap_client.proto";
import "ap_mesh.proto";
import "ap_rogue.proto";
import "sci-event.proto";
import "sci-configuration.proto";
import "ap_avc.proto";
import "ap_avc_all.proto";
import "sci-alarm.proto";
import "ap_wired_client.proto";
import "ap_hccd_report.proto";
import "sci-pci.proto";
import "switch_all.proto";
import "switches.proto";
import "sci-rogue.proto";
import "ap_peerlist.proto";
import "session_manager.proto";

message SciMessage {
    extensions 1001 to 3000;

    //protocol version
    optional string version = 1;
 optional bytes uuid = 3;
 optional int64 sentTimeMs = 4;
 optional string sciSystemId = 5;
 optional APStatus apStatus = 101;
 optional APReportStats apReport = 102;
 optional APClientStats apClient = 103;
 optional APMeshStats apMesh = 104;
 optional RogueAPStats apRogue = 105;
 optional EventMessage eventMessage = 106;
 optional ConfigurationMessage configurationMessage = 107;
 optional AlarmMessage alarmMessage = 108;
 optional APWiredClientStats apWiredClient = 109;
 optional PciReportMessage pciReportMessage = 110;
 optional ApHccdReportMessage apHccdReportMessage = 111;
 optional com.ruckuswireless.scg.protobuf.icx.SwitchMessage switchMessage = 112;
 optional SciRogueMessage sciRogueMessage = 113;
 optional SessionManagerClientData sessionManagerClientData = 114;
 repeated ArcMessage arcMessage = 206;
 optional APAVCStats apAvc = 207;
 optional com.ruckuswireless.scg.protobuf.icx.SwitchConfigurationMessage switchConfigurationMessage =
300;
 optional com.ruckuswireless.scg.protobuf.icx.RealtimeSwitchStatus realtimeSwitchStatus = 301;
 optional APPeerReport apPeer = 302;
 optional com.ruckuswireless.scg.protobuf.icx.SwitchDetailMessage switchDetailMessage = 303;

}
```

Field Description

TABLE 74 SCI Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	string	snapshot	NULL	protocol version.
uuid	bytes	snapshot	NULL	Message identifier for duplication detection.
sentTimeMs	int64	snapshot	NULL	Sent time in milliseconds.
sciSystemId	string	snapshot	NULL	SCI System identifier.
apStatus	.APStatus	snapshot	NULL	APStatus from Routine AP Statistic Report.
apReport	.APReportStats	snapshot	NULL	APReportStats from Routine AP Statistic Report.
apClient	.APClientStats	snapshot	NULL	APClientStats from Routine AP Statistic Report.
apMesh	.APMeshStats	snapshot	NULL	APMeshStats from Routine AP Statistic Report.
apRogue	.RogueAPStats	snapshot	NULL	RogueAPStats from Routine AP Statistic Report.
eventMessage	.EventMessage	snapshot	NULL	Event message which has been applied to be sent to outer service.
configurationMessage	.ConfigurationMessage	snapshot	NULL	SZ overall configuration message sent out every 15 minutes.
alarmMessage	.AlarmMessage	snapshot	NULL	Alarm message will be sent out when the alarm happens.
apWiredClient	.APWiredClientStats	snapshot	NULL	APWiredClientStats from Routine AP Statistic Report.
pciReportMessage	.PciReportMessage	snapshot	NULL	PCI Compliance Report which will be sent out every 15 minutes.
apHccdReportMessage	.ApHccdReportMessage	snapshot	NULL	ApHccdReportMessage from Routine AP Statistic Report.
switchMessage	.com.ruckuswireless.scg.protobuf.icx.SwitchMessage	snapshot	NULL	SwitchMessage from ICX Statistic Report.
sciRogueMessage	.SciRogueMessage	snapshot	NULL	SciRogueMessage from AP rogue AP report which have been classified by rogue AP policy of SZ.
sessionManagerClientData	SessionManagerClientData	snapshot	NULL	SessionManagerClientData from Session Manager's client auth/ deauth or connect/disconnect event.
arcMessage	.ArcMessage	snapshot	NULL	ArcMessage is from Routine AP Statistic Report which will be sent out every 5 minutes.
apAvc	.APAVCStats	snapshot	NULL	APAVCStats contains a series of ArcMessage.
switchConfigurationMessage	.com.ruckuswireless.scg.protobuf.icx.SwitchConfigurationMessage	snapshot	NULL	Switch overall configuration message sent out every 15 minutes.

TABLE 74 SCI Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
realtimeSwitchStatus	.com.ruckuswireless.scg.protobuf.icx.RealtimeSwitchStatus	snapshot	NULL	Switch realtime status.
apPeer	APPeerReport	snapshot	NULL	APPeerReport for on-demand API.
switchDetailMessage	com.ruckuswireless.scg.protobuf.icx.SwitchDetailMessage	snapshot	NULL	The list of attached switch detail

sci-pci.proto

```
* Copyright 2017 Ruckus Wireless, Inc. All rights reserved.
*
* RUCKUS WIRELESS, INC. CONFIDENTIAL -
* This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is
* fully protected under copyright and trade secret laws. You may not view,
* use, disclose, copy, or distribute this file or any information contained
* herein except pursuant to a valid license from Ruckus.
*
*    JsonMessage GPB format is used to transfer the JSON messages across
*    applications which includes the version, message content and message
*    content type attributes.
*/
option java_package = "com.ruckuswireless.scg.protobuf.sci";
message PciReportMessage {
    optional uint32 version = 1;
    optional string pciReportUuid = 2;
    optional bool changedPassword = 3 [default = true];
    repeated ControllerSummaryMessage controllerSummary = 4;
    optional bool enabledSSL = 5 [default = true];
    optional bool blockedTelnet = 6 [default = true];
    optional bool enabledPasswordAuthentication = 7 [default = true];
    optional bool encryptedCredential = 8 [default = true];
    optional bool enabledPasswordStandard= 9 [default = true];
    optional bool enabledUniquePassword= 10 [default = false];
    optional bool disabledGenericAccount = 11 [default = false];
    optional bool synchronizeNTP = 12 [default = true];
    optional bool alertConfigurationChange = 13 [default = true];
    repeated AccountSecurityMessage accountSecurity = 14;
    repeated WlanInformationMessage wlanInformation = 15;
}

message AccountSecurityMessage {
    optional string userUuid = 1;
    optional string userName = 2;
    optional string accountSecurityUuid = 3;
    optional string accountSecurityName = 4;
    optional string description = 5;
    optional string domainId = 6;
    optional uint32 accountLockout = 7;
    optional uint32 lockoutDuration = 8;
    optional uint32 passwordExpiration = 9;
    optional uint32 passwordReuse = 10;
    optional uint32 sessionIdle = 11;
    optional bool twoFactorAuthEnabled = 12;
    optional uint32 disableInactiveAccounts = 13;
}

message ControllerSummaryMessage {
    optional string controllerSummaryUuid = 1;
    optional string modelName = 2;
}

message WlanInformationMessage {
    optional string id = 1;
    optional string zoneId = 2;
    optional string wlanName = 3;
    optional string ssid = 4;
    optional string vlanId = 5;
    optional string securityMethod = 6;
    optional string wpaVersion = 7;
}
```

Field Description

TABLE 75 PCI Report Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
pciReportUuid	string	snapshot	NULL	PCI report UUID
changedPassword	boolean	snapshot	NULL	Default password of SZ is changed or not
controllerSummary	.ControllerSummaryMessage	Serialization	NULL	Controller Summary
enabledSSL	boolean	snapshot	NULL	SSL of SZ web is enabled or not
blockedTelnet	boolean	snapshot	NULL	Telnet of SZ is blocked or not
enabledPasswordAuthentication	boolean	snapshot	NULL	Authentication mechanism is enabled on each user of SZ or not
encryptedCredential	boolean	snapshot	NULL	Credentials of SZ are encrypted or not
enabledPasswordStandard	boolean	snapshot	NULL	Password standards of SZ are enabled or not
enabledUniquePassword	boolean	snapshot	NULL	Unique password of SZ is enabled or not
disabledGenericAccount	boolean	snapshot	NULL	Generic accounts of SZ are disabled or not
synchronizeNTP	boolean	snapshot	NULL	Time of SZ is synchronized to NTP or not
alertConfigurationChange	boolean	snapshot	NULL	Configuration change alert of SZ is enabled or not
accountSecurity	.AccountSecurityMessage	Serialization	NULL	Account Security
wlanInformation	.WlanInformationMessage	Serialization	NULL	Wlan information messages

TABLE 76 PCI Account Security Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
userUuid	string	snapshot	NULL	User UUID
userName	string	snapshot	NULL	User name
accountSecurityUuid	string	snapshot	NULL	Account security UUID
accountSecurityName	string	snapshot	NULL	Account security name
description	string	snapshot	NULL	Account security description
domainId	string	snapshot	NULL	Domain ID
accountLockout	uint32	snapshot	NULL	Failed authentication attempts before account lockout
lockoutDuration	uint32	snapshot	NULL	The duration for which the account is automatically locked without administrative intervention
passwordExpiration	uint32	snapshot	NULL	Time of password expiration

TABLE 76 PCI Account Security Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
passwordReuse	uint32	snapshot	NULL	Password reuse setting
sessionIdle	uint32	snapshot	NULL	Session idle timeout
twoFactorAuthEnabled	boolean	snapshot	NULL	Two-Factor authentication is enabled or not
disableInactiveAccounts	uint32	snapshot	NULL	Time of disabling inactive account

TABLE 77 PCI Control Summary Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
controllerSummaryUuid	string	snapshot	NULL	Controller summary UUID
modelName	string	snapshot	NULL	Model name

TABLE 78 PCI WLAN Information Message

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
Id	string	snapshot	NULL	WLAN ID
ZoneID	string	snapshot	NULL	Zone UUID
WlanName	string	snapshot	NULL	WLAN name
ssid	string	snapshot	NULL	SSID
vlanId	string	snapshot	NULL	Vlan ID
securityMethod	string	snapshot	NULL	WLAN security method
wpaVersion	string	snapshot	NULL	WPA version

sci-rogue.proto

```
syntax = "proto2";
option java_package = "com.ruckuswireless.scg.protobuf.sci";

import "ap_rogue.proto";

/**
 * @internal
 **/
message SciRogueMessage {
    optional RogueAPStats apRogue = 1;
    repeated RoguePolicyMessage roguePolicies = 2;
}


/**
 * @internal
 **/
message RoguePolicyMessage {
    optional string rogueMac = 1;
    optional string policyName = 2;
    optional string ruleName = 3;
    optional string type = 4;
}
```

session_manager.proto

```
/**
 * Copyright 2018 Ruckus Wireless, Inc. All rights reserved.
 *
 * RUCKUS WIRELESS, INC. CONFIDENTIAL -
 * This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is
 * fully protected under copyright and trade secret laws. You may not view,
 * use, disclose, copy, or distribute this file or any information contained
 * herein except pursuant to a valid license from Ruckus.
 */
option java_package = "com.ruckuswireless.scg.protobuf.client";

import "ScgSessMgrPubIpc.proto";

message SessionManagerClientData {
    optional ScgSessMgrPubIpc.GBP_SESSMGR_PUBS_STATS_AP_CLIENT_STAT_T sessionMgrClientStat = 1;
    optional ExtraSessClientDataAttributes extraAttributes = 2;
}

message ExtraSessClientDataAttributes {
    optional string serialNumber = 1;
    optional string domainName = 2;
    optional string zoneName = 3;
    optional string apGroupId = 4;
    optional string apGroupName = 5;
    optional string wlanName = 6;
}
```

simple-storage.proto

```
/syntax = "proto2";
import "google/protobuf/descriptor.proto";
package com.ruckuswireless.scg.protobuf.storage;

/*
 * The messages defined in this file should keep unknown to outsider.
 */

enum Category {
    // Config category will be persisted and index-ed.
    CONFIG = 1;
    // Status category will be index-ed but not persisted.
    STATUS = 2;
    // Device were a read-only index reflects all contents of enclosed Status or Config.
    DEVICE = 3;
    // Statistic will be index-ed and rotated according to SZ profile.
    STATISTIC = 4;
    // File
    FILE = 5;
    // Data will be persisted and not index-ed.
    DATA = 6;
    // For tree structured data.
    GROUP = 7;
    // Information category only indexed.
    INFORMATION = 8;
}

extend google.protobuf.FileOptions {
    optional int64 schemaVersion = 50000;
}

extend google.protobuf.MessageOptions {
    optional Category category = 50000;
    optional bool entity = 50001;
}

extend google.protobuf.FieldOptions {
    optional bool id = 50000;
    optional bool version = 50001;
    optional string ingest = 50002;
    optional bool fileContent = 50003;
}
```

switch_all.proto

```
*Copyright 2013 Ruckus Wireless, Inc. All rights reserved.
*
*    RUCKUS WIRELESS, INC. CONFIDENTIAL -
*    This is an unpublished, proprietary work of Ruckus Wireless, Inc., and is fully protected under
copyright and trade secret laws. You may not view, use, disclose, copy, or distribute this file or any
information contained herein except pursuant to a valid license from Ruckus.
*
*    JsonMessage GPB format is used to transfer the JSON messages across
*    applications which includes the version, message content and message
*    content type attributes.
*/
option java_package = "com.ruckuswireless.scg.protobuf";
message SwitchMessage {
    required uint32 version = 1;
    optional .com.ruckuswireless.scg.protobuf.icx.SwitchStatus switchStatus = 2;
    optional .com.ruckuswireless.scg.protobuf.icx.SwitchStats switchStats = 3;
    repeated .com.ruckuswireless.scg.protobuf.icx.PortStatus portStatuses = 4;
    repeated .com.ruckuswireless.scg.protobuf.icx.PortStats portStats = 5;
    repeated .com.ruckuswireless.scg.protobuf.icx.ConnectedDeviceStatus connectedDevicesStatus = 6;
    repeated .com.ruckuswireless.scg.protobuf.icx.SwitchUnitStatus switchUnitStatuses = 7;
               repeated SwitchClientVisibility switchClientVisibility = 8;
    optional SwitchClientStatus switchClientStatus = 9;
    repeated LAGStatus lagStatuses = 10;
}
```

Field Description

TABLE 79 Switch Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
version	uint32	snapshot	NULL	
switchStatus	.com.ruckuswireless.scg.protobuf.icx.SwitchStatus	snapshot	NULL	Status of switch, ex: CPU, Memory, System Information, System Network Information.
switchStats	.com.ruckuswireless.scg.protobuf.icx.SwitchStats	snapshot	NULL	Stats of switch for statistical analysis, ex Network Traffic, CPU, Memory.
portStatuses	.com.ruckuswireless.scg.protobuf.icx.PortStatus	snapshot	NULL	Status of port, ex: CPU, Memory, Port Information(Network, PoE, Traffic, Packets In/Out).
portStats	.com.ruckuswireless.scg.protobuf.icx.PortStats	snapshot	NULL	Stats of port for statistical analysis, ex: Port Network Traffic.
connectedDevicesStatus	.com.ruckuswireless.scg.protobuf.icx.ConnectedDeviceStatus	snapshot	NULL	Status of connected device, ex: Remote Port Information(Device Name, Mac Address, Type, Description), Local Port Information(Name, Mac Address).
switchUnitStatuses	.com.ruckuswireless.scg.protobuf.icx.SwitchUnitStatus	snapshot	NULL	Status of switch unit, ex: Switch Unit Information(ID, Uptime, Status, Serial Number).
switchClientVisibility	.com.ruckuswireless.scg.protobuf.icx.SwitchClientVisibility	snapshot	NULL	Visibility of switch client
switchClientStatus	SwitchClientStatus	snapshot	NULL	Status of switch client

TABLE 79 Switch Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
lagStatuses	.com.ruckuswireless.scg.protobuf.icx.LAGStatus	snapshot	NULL	Status of LAG port

switches.proto

```
syntax = "proto2";

import "simple-storage.proto";
import "commons.proto";

package com.ruckuswireless.scg.protobuf.icx;

/**
 * @internal
 **/
enum CliTask {
    NONE = 1;
    RELOAD = 2;
    DELETE = 3;
    DISCONNECT = 4;
}

/**
 * @internal
 **/
enum RegistrationStatus {
    UNKNOWN = 1;
    REGISTRATION_PENDING = 2;
    PENDING = 3;
    OVERLOADING = 4;
    APPROVED = 5;
    REJECTED = 6;
}

/**
 * @internal
 **/
message GpbNetworkSwitch {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = DEVICE;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Switch/Stack Router
ID + MAC
    optional SwitchConfig switchConfig = 2;
    optional SwitchStatus switchStatus = 3;
}

/**
 * @internal
 **/
message SwitchConfig {

    enum ProvisionState {
        INIT = 1;
        FAILURE = 2;
        SUCCESS = 3;
    }

    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string hostName = 2;
    optional RegistrationStatus registrationStatus = 3;
    optional string activeIpList = 4;
    optional string passiveIpList = 5;
    optional string sshAccountName = 6;
    optional string switchRsaPublicKey = 7;
    optional string switchRsaPrivateKey = 8;
    optional string szSshPublicKey = 9;
    optional string syslogIpAddress = 10;
    optional int32 syslogPort = 11;
    optional string snmpIpAddress = 12;
    optional int32 snmpPort = 13;
    optional string cliIpAddress = 14;
    optional int32 cliPort = 15;
```

```
    optional string domainId = 16;
    optional string tenantId = 17;
    optional string switchGroupLevelOneId = 18;
    optional string switchGroupLevelTwoId = 19;
    optional string switchSshKey = 20;
    optional string serialNumber = 21;
    optional string routerId = 22;
    optional string szNodeAttached = 23;
    optional CliTask cliTask = 24 [default = NONE];
    optional string macAddress = 26;
    optional int64 lastBackupTime = 27;
    optional int64 lastRestoreTime = 28;
    optional string lastBackupStatus = 29;
    optional string lastRestoreStatus = 30;
    optional string ipAddress = 31;
    optional int64 createdTimestamp = 32;
    optional int64 lastConfigTime = 33;
    optional int32 szHttpClientPort = 34;
    optional int32 supportedCsl = 35;
    optional bool cacheFactoryDefault = 36;
    optional bool isCslRequested = 37;
    optional string cloudPort = 38;
    optional string licenseType = 39;
    optional string firmwareUpdateScheduleId = 40;
    optional string firmwareUpdateScheduledTime = 41;
    optional string firmwareUpdateModifiedTime = 42;
    optional string firmwareUpdateStatus = 43;
    optional string firmwareUpdateToVersion = 44;
    optional bool isMigration = 45;
    optional ProvisionState provisionState = 46;
    optional bool operational = 47;
}

/**
 * @internal
 */

message PowerSupplyGroup {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string stackId = 2;
    optional int32 powerSlotNum = 3;
    optional string powerSupplyType = 4;
    optional string powerSupplyStatus = 5;
    optional string serialNumber = 6;
}

message TempThresholdGroup {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string stackId = 2;
    optional int32 slotNum = 3;
    optional int32 tempThreshold = 4;
    optional double temperatureValue = 5;
}

message FanGroup {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string stackId = 2;
    optional int32 slotNum = 3;
    optional string type = 4;
    optional string status = 5;
    optional string serialNumber = 6;
}

message SwitchStatus {
    required string id = 1;
    optional string firmware = 2;
    optional string model = 3;
    optional string defaultGateway = 4;
    optional int32 numOfPorts = 5;
```

```
    optional string natIp = 6;
    optional string status = 7;
    optional bool poeAvailable = 8;
    optional uint64 lastBackup = 9;
    optional uint64 cpu = 10;
    optional uint64 memory = 11;
    optional string uptime = 12;
    optional int32 alerts = 13;
    optional bool isStack = 14;
    optional string stackId = 15;
    optional int32 priority = 16;
    optional string modules = 17;
    optional string domainId = 18;
    optional string groupId = 19;
    optional uint64 lastUpdateTimeInMillis = 20;
    optional string switchName = 21;
    optional string tenantId = 22;
    optional string switchGroupLevelOneId = 23;
    optional string switchGroupLevelTwoId = 24;
    optional string serialNumber = 25;
    optional string unitSerialNumbers = 26;
    optional string portModuleIds = 27;
    optional string partitionInUse = 28;
    optional string switchMode = 29;
    optional string switchSWVersion = 30;
    optional int32 numOfUnits = 31;
    optional int32 poeUtilization = 32;
    optional int32 poeTotal = 33;
    optional int32 poeFree = 34;
    optional string family = 35;
    optional string cloudPort = 43;
    optional string domainName = 44;
    optional string switchGroupLevelOneName = 45;
    optional string switchGroupLevelTwoName = 46;
    optional string powerSupplyGroups = 47;
    optional string fanGroups = 48;
    optional string ipAddress = 49;
    optional string subnetMask = 50;
    optional string staticOrDynamic = 51;
    optional string dns = 52;
    optional string temperatureGroups = 53;
    optional uint64 totalMemory = 54;
    optional uint64 freeMemory = 55;
    optional bool hasPoECapability = 56;
    optional bool operational = 57;
    optional string reservedField1 = 58;
    optional string reservedField2 = 59;
    optional string reservedField3 = 60;
    optional string reservedField4 = 61;
    optional LocalSyncRunningStatus localsyncStatus = 62;
}

message SwitchStats {
    required string id = 1;
    optional int64 timestamp = 2;
    optional string switchSerialNum = 3;
    optional int64 tx = 4;
    optional int64 rx = 5;
    optional int64 txRx = 6;
    optional int64 txPkt = 7;
    optional int64 rxPkt = 8;
    optional int64 txRxPkt = 9;
    optional int64 txRate = 10;
    optional int64 rxRate = 11;
    optional int64 txRxRate = 12;
    optional int64 cpu = 13;
    optional int64 memory = 14;
    optional string domainId = 15;
    optional string tenantId = 16;
    optional string switchGroupLevelOneId = 17;
    optional string switchGroupLevelTwoId = 18;
    optional string domainName = 19;
```

```
    optional string switchGroupLevelOneName = 20;
    optional string switchGroupLevelTwoName = 21;
    optional int64 multicastOut = 22;
    optional int64 multicastIn = 23;
    optional int64 broadcastOut = 24;
    optional int64 broadcastIn = 25;
    optional int64 crcErr = 26;
    optional int64 outErr = 27;
    optional int64 inErr = 28;
    optional int64 unicastOut = 29;
    optional int64 unicastIn = 30;
    optional bool collectorExecuted = 31;
}

message PortStats {
    required string id = 1;
    optional int64 timestamp = 2;
    optional string switchSerialNum = 3;
    optional string portMac = 4;
    optional int64 tx = 5;
    optional int64 rx = 6;
    optional int64 txRx = 7;
    optional int64 txPkt = 8;
    optional int64 rxPkt = 9;
    optional int64 txRxPkt = 10;
    optional int64 txRate = 11;
    optional int64 rxRate = 12;
    optional int64 txRxRate = 13;
    optional string domainId = 14;
    optional string tenantId = 15;
    optional string switchGroupLevelOneId = 16;
    optional string switchGroupLevelTwoId = 17;
    optional string switchId = 18;
    optional string domainName = 19;
    optional string switchGroupLevelOneName = 20;
    optional string switchGroupLevelTwoName = 21;
    optional string switchUnitId = 22;
}

message SwitchGroup {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string domainId = 2;
    optional string creatorId = 3;
    optional string tenantId = 4;
    optional string switchGroupLevelOneId = 5;
    optional string switchGroupLevelTwoId = 6;
    optional uint64 createDatetime = 7;
    optional string name = 8;
    optional string description = 9;
    optional string firmware = 10;
}

message GpbNetworkPort {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = DEVICE;
    optional string portMac = 1;
    optional string switchSerialNum = 2;
    optional string switchUnitId = 3;
    optional PortConfig portConfig = 4;
    optional PortStatus portStatus = 5;
    optional string portId = 6 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //
portMac_portIdentifier
}

message PortConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;
    optional string portMac = 1;
    optional string switchId = 2;
    optional string switchUnitId = 3;
    optional string portId = 4 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
}
```

```
message PortStatus {
    optional string portMac = 1;
    optional string switchId = 2;
    optional string type = 3;
    optional string name = 4;
    optional string status = 5;
    optional string adminStatus = 6;
    optional string vlanIds = 7;
    optional string neighborName = 8;
    optional string portSpeed = 9;
    optional bool ruckusDevice = 10;
    optional bool lldpEnabled = 11;
    optional string lagName = 12;
    optional string lagStatus = 13;
    optional string spanningTreeStatus = 14;
    optional int32 poeUsed = 15;
    optional int32 poeTotal = 16;
    optional double poePercent = 17;
    optional string domainId = 18;
    optional string switchGroupLevelOneId = 19;
    optional int64 tx = 20;
    optional int64 rx = 21;
    optional double signalIn = 22;
    optional double signalOut = 23;
    optional int64 crcErr = 24;
    optional int64 inErr = 25;
    optional int64 outErr = 26;
    optional string opticsType = 27;
    optional string tenantId = 29;
    optional string switchGroupLevelTwoId = 30;
    optional string portSpeedCapacity = 31;
    optional string switchUnitId = 32;
    optional string portIfaceName = 33;
    optional bool poeEnabled = 34;
    optional bool usedInFormingStack = 35;
    optional string portIdentifier = 36;
    optional string unTaggedVlan = 37;
    optional bool isInWarningState = 38;
    optional int64 inDiscard = 39;
    optional int64 broadcastIn = 40;
    optional int64 broadcastOut = 41;
    optional int64 multicastIn = 42;
    optional int64 multicastOut = 43;
    optional string poeType = 44;
    optional string portIdentifierFormatted = 45;
    optional string portId = 46 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string domainName = 47;
    optional string switchGroupLevelOneName = 48;
    optional string switchGroupLevelTwoName = 49;
    optional int64 unicastOut = 50 [(com.ruckuswireless.scg.protobuf.storage.ingest) = "delta"];
    optional int64 unicastIn = 51 [(com.ruckuswireless.scg.protobuf.storage.ingest) = "delta"];
    optional string vlanDetailInformation = 52;
    optional string switchName = 53;
    optional string neighborMacAddress = 54;
    optional string inAclConfigName = 55;
    optional string outAclConfigName = 56;
    optional int32 poeClass = 57;
    optional int32 poePriority = 58;
    optional int32 poeBudget = 59;
    optional int32 lagId = 60;
    optional string lagAdminStatus = 61;
    optional string searchableVlans = 62;
    optional PortType resourceName = 63;
    optional bool isLagMember = 64;
    optional string poePdClass = 65;
    optional string poePdClassB = 66;
    optional string poeLldpMaxPowerRequest = 67;
    optional string poeLldpMaxPowerRequestA = 68;
    optional string poeLldpMaxPowerRequestB = 69;
    optional PoePortCapability poePortCapability = 70;
    optional string poe2PairMaxPower = 71;
```

```
    optional string poe4PairMaxPower = 72;
    optional PoeOverDriveMode poeOverdriveMode = 73;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated GpbNetworkPort ports = 1000;
}

message Client {
    option (com.ruckuswireless.scg.protobuf.storage.category) = STATUS;
    optional string mac = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string switchSerialNum = 2;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated GpbNetworkSwitch switches = 1001;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SwitchGroup switchGroups = 1002;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated RegistrationRule registrationRules = 1003;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated FanGroup fanGroups = 1004;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated TempThresholdGroup tempThresholdGroups = 1005;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated PowerSupplyGroup powerSupplyGroups = 1006;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated GpbSwitchUnit switchUnits = 1007;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated Job jobs = 1008;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated GpbConnectedDevice connectedDevices = 1009;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SwitchCredential switchCredentials = 1010;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated ConfigBackup configBackups = 1011;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated Schedule schedules = 1012;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SystemConfig systemConfigs = 1013;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated GroupModelConfig groupModelConfigs = 1014;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated VlanConfig vlanConfigs = 1015;
```

```
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated AaaServer aaaServers = 1016;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated DeployLog deployLogs = 1017;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated DeployLogItem deployLogItems = 1018;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SwitchConfigStore switchConfigStores = 1019;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated CommonSettings commonSettings = 1020;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated StaticRoute staticRoutes = 1021;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated ACLConfig aclConfigs = 1022;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SpecificSettings specificSettings = 1023;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated PortConfiguration portConfiguration = 1024;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated VEConfig veConfigs = 1025;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated LAGConfig lagConfigs = 1027;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated StackConfig stackConfigs = 1028;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated VlanPortRelation vlanPortRelations = 1029;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated DeploySchedule deploySchedules = 1030;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated DeployPending deployPendings = 1031;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SwitchClientVisibility switchClientVisibilities = 1032;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated SupportSaveStatus supportSaveStatus = 1033;
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated KumoSwitch kumoSwitch = 1034;
```

```
}

extend com.ruckuswireless.scg.protobuf.storage.ListModel {
    repeated AAASetting aaaSetting = 1035;
}

message Job {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string type = 2;
    optional string action = 3;
    optional string status = 4;
    optional string failureReason = 5;
    optional string switchId = 6;
    optional string domainId = 7;
    optional string tenantId = 8;
    optional string switchGroupLevelOneId = 9;
    optional string switchGroupLevelTwoId = 10;
    optional int64 createdTimestamp = 11;
    optional int64 modifiedTimestamp = 12;
    optional string csvData = 13;
    optional string stickyNodeId = 14;
    optional string scheduleId = 15;
    optional int32 retryCount = 16;
}

message Schedule {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string type = 2;
    optional string triggerValue = 3;
    optional int64 createdTimestamp = 4;
    optional string switchId = 5;
    optional string jobType = 6;
    optional string jobAction = 7;
    optional string csvData = 8;
    optional string stickyNodeId = 9;
    optional string status = 10;
}

enum RegistrationRuleType {
    IP_RANGE = 1;
    SUBNET = 2;
    MODEL_NUMBER = 3;
}

message RegistrationRule {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    // No suitable category, set the category to GROUP for now.
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string creatorId = 2;
    optional string switchGroupId = 3;
    optional RegistrationRuleType type = 4;
    optional string ipFrom = 5;
    optional string ipTo = 6;
    optional string network = 7;
    optional string subnetMask = 8;
    optional string modelNumber = 9;
    optional int32 rank = 10;
    optional uint64 createDatetime = 11;
    optional string description = 12;
}

message GpbSwitchUnit {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = DEVICE;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional SwitchUnitConfig switchUnitConfig = 2;
    optional SwitchUnitStatus switchUnitStatus = 3;
```

```
}

message SwitchUnitConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Switch-Unit serial
number
    optional string switchId = 2; //Reference key for Switch/Stack
    optional string model = 3;
    optional string macAddress = 4;
    optional int32 numOfPorts = 5;
    optional string modules = 6;
    optional string unitName = 7;
    optional string unitSlNum = 8;
    optional string portStatusUp = 9;
    optional string portStatusWarning = 10;
    optional string portStatusDown = 11;
}

message  SwitchUnitStatus {
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Switch-Unit serial
number
    optional string switchId = 2;
    optional string upTime = 3;
    optional string unitStatus = 4; //Active Or Passive
    optional string unitSlNum = 8;
    optional string domainId = 9;
    optional string switchGroupLevelOneId = 10;
    optional string switchGroupLevelTwoId = 11;
    optional string domainName = 12;
    optional string switchGroupLevelOneName = 13;
    optional string switchGroupLevelTwoName = 14;
    optional int32 unitId = 15;
    optional string unitState = 16;
    optional string unitName = 17;
    optional int32 poeUtilization = 18;
    optional int32 poeTotal = 19;
    optional int32 poeFree = 20;
}

message GpbConnectedDevice {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = DEVICE;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Remote Port MAC
    optional ConnectedDeviceConfig connectedDeviceConfig = 2;
    optional ConnectedDeviceStatus connectedDeviceStatus = 3;
}

message ConnectedDeviceConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Remote Port MAC
    optional string remotePortMac = 2; //Remote Port MAC
    optional string localPortMac = 3;
    optional string switchId = 4;
    optional string remoteDeviceMac = 5;
}

message ConnectedDeviceStatus {
    option (com.ruckuswireless.scg.protobuf.storage.category) = STATUS;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //Remote Port MAC
    optional string remotePortMac = 2;
    optional string remoteDeviceName = 3;
    optional string remotePortType = 4;
    optional string remotePortDesc = 5;
    optional string localPort = 6;
    optional string remotePort = 7;
    optional string isRuckusAP = 8;
    optional string domainId = 9;
    optional string tenantId = 10;
    optional string switchGroupLevelOneId = 11;
    optional string switchGroupLevelTwoId = 12;
    optional string switchId = 13;
    optional string unitId = 14;
```

```
    optional string localPortIfaceName = 15;
    optional string localPortMac = 16;
    optional string domainName = 17;
    optional string switchGroupLevelOneName = 18;
    optional string switchGroupLevelTwoName = 19;
    optional int64 updatedTime = 20;
    optional string remoteDeviceMac = 21;
}

message LAGStatus {
    option (com.ruckuswireless.scg.protobuf.storage.category) = STATUS;
    optional string lagName = 1;
    optional int32 lagId = 2;
    optional string lagStatus = 3;
    optional string lagAdminStatus = 4;
    optional string vlanIds = 5;
    optional string switchId = 6;
    optional string unTaggedVlan = 7;
    optional string switchUnitId = 8;
    optional string domainName = 9;
    optional string switchGroupLevelOneName = 10;
    optional PortType resourceName = 11;
    optional bool isLagMember = 12;
}

enum SwitchCredentialSNMPType {
    SNMP_V2 = 1;
    SNMP_V3 = 2;
}

enum SwitchCredentialAuthProtocolType {
    SHA = 1;
    MD5 = 2;
}

enum SwitchCredentialPrivacyProtocolType {
    PROTO_NONE = 1;
    AES = 2;
    DES = 3;
}

message SwitchCredential {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string creatorName = 2;
    optional SwitchCredentialSNMPType type = 3;
    optional string description = 4;
    optional string community = 5;
    optional string userName = 6;
    optional string authPassword = 7;
    optional SwitchCredentialAuthProtocolType authProtocol = 8;
    optional string privacyPassword = 9;
    optional SwitchCredentialPrivacyProtocolType privacyProtocol = 10;
    optional int32 rank = 11;
    optional uint64 createDatetime = 12;
}

message ConfigBackup {
    option (com.ruckuswireless.scg.protobuf.storage.entity) = true;
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    enum BackupType {
        MANUAL = 1;
        SCHEDULED = 2;
    }
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string name = 2;
    optional int64 backupStartTime = 3;
    optional int64 backupEndTime = 4;
    optional int64 lastRestoreStartTime = 5;
    optional int64 lastRestoreEndTime = 6;
    optional BackupType type = 7;
```

```
    optional string switchId = 8;
    optional string config = 9;
    optional string backupStatus = 10;
    optional string lastRestoreStatus = 11;
    optional string domainId = 12;
    optional string switchGroupLevelOneId = 13;
    optional string switchGroupLevelTwoId = 14;
    optional string failureReason = 15;
    optional int64 backupStartTriggerTime = 16;
    optional int64 restoreStartTriggerTime = 17;
    optional int32 retryCount = 18;
}

message SystemConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = DATA;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string scepEnrollments = 2;
    optional string eventConfig = 3;
    optional string switchTrackConfig = 4;
    optional string aaaSettings = 5;
    optional int64 postUpgradeTime = 6;
}

message GroupModelConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string familyId = 2;
    optional string groupId = 3;
    optional int64 createdTime = 4;
    optional int64 updatedTime = 5;
    optional bool hasVlanConfig = 6;
    optional bool hasAclConfig = 7;
    optional bool hasStaticRouteConfig = 8;
    optional bool hasConfigured = 9;
    optional bool hasSelected = 10;
}

message VlanConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;


    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional int32 vlanId = 2;
    optional string name = 3;
    optional string groupId = 4;
    optional string familyId = 5;
    optional string switchId = 6;
    optional bool enableIpv4DhcpSnooping = 7;
    optional string ipv4DhcpSnoopingTrustPort = 8;
    optional bool enableArpInspection = 9;
    optional string arpInspectionTrustPort = 10;
    optional IgmpSnoopingType igmpSnooping = 11;
    optional string multicastVersion = 12;
    optional SpanningTreeType spanningTree = 13;
    optional int32 spanningTreePriority = 14;
    optional string portsConfig = 15;
    optional PushTimeType pushTimeType = 16;
    optional int64 pushTime = 17;
    optional int64 createdTime = 18;
    optional int64 updatedTime = 19;
    optional string arpInspectionsConfig = 20;
    optional int64 scheduled = 21;
    optional bool initialConfig = 22;
    optional string rootBridgeFamilyId = 23;

}

enum Level {
    READ_WRITE = 1;
    PORT_CONFIG = 2;
```

```
    READ_ONLY = 3;
    LEVEL_NONE = 4;
}

enum ServerType {
    RADIUS = 1;
    TACACS_PLUS = 2;
    LOCAL = 3;
    SERVER_TYPE_NONE = 4;
}

message AAASetting {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];

    optional bool authnEnabledSSH = 2;
    optional bool authnEnableTelnet = 3;
    optional ServerType authnFirstPref = 4 [default = SERVER_TYPE_NONE];
    optional ServerType authnSecondPref = 5 [default = SERVER_TYPE_NONE];
    optional ServerType authnThirdPref = 6 [default = SERVER_TYPE_NONE];

    optional bool authzEnabledCommand = 7;
    optional bool authzEnabledExec = 8;
    optional Level authzCommonsLevel = 9 [default = LEVEL_NONE];
    optional ServerType authzCommonsServer1 = 10 [default = SERVER_TYPE_NONE];
    optional ServerType authzCommonsServer2 = 11 [default = SERVER_TYPE_NONE];
    optional ServerType authzExecServer1 = 12 [default = SERVER_TYPE_NONE];
    optional ServerType authzExecServer2 = 13 [default = SERVER_TYPE_NONE];

    optional bool acctEnabledCommand = 14;
    optional bool acctEnabledExec = 15;
    optional Level acctCommonsLevel = 16 [default = LEVEL_NONE];
    optional ServerType acctCommonsServer1 = 17 [default = SERVER_TYPE_NONE];
    optional ServerType acctCommonsServer2 = 18 [default = SERVER_TYPE_NONE];
    optional ServerType acctExecServer1 = 19 [default = SERVER_TYPE_NONE];
    optional ServerType acctExecServer2 = 20 [default = SERVER_TYPE_NONE];
    optional int64 updatedTime = 21;
}

message AaaServer {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    /**
     * @internal
     **/
    enum Purpose {
        DEFAULT = 1;
        AUTHENTICATION_ONLY = 2;
        AUTHORIZATION_ONLY = 3;
        ACCOUNTING_ONLY = 4;
    }

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string name = 2;
    optional int64 createdTime = 3;
    optional int64 updatedTime = 4;
    optional ServerType serverType = 5;

    // radius and tacacs
    optional string ip = 6;
    optional int32 authPort = 7;
    optional int32 acctPort = 8;
    optional string secret = 9;

    // tacacs
    optional Purpose purpose = 10;

    // local user
    optional string username = 11;
    optional string password = 12;
    optional Level level = 13;
```

```
    // creator & updater
    optional string creatorUsername = 14;
    optional string creatorId = 15;
    optional string updaterUsername = 16;
    optional string updaterId = 17;
    optional string groupId = 18;
}

enum DeployScope {
    PRE_PROVISION = 1;
    PROVISION = 2;
       GROUP = 3;
    PORT = 4;
    SWITCH = 5;
       COPY = 6;
       OVERWRITE = 7;
}

enum ConfigFeature {
    VLAN = 1;
    ACL = 2;
    STATIC_ROUTE = 3;
}

enum ConfigType {
    PROVISIONING = 1;
    GLOBAL = 2;
    COMMON = 3;
    MODEL = 4;
    SWITCH_SETTINGS = 5;
    PORT_SETTINGS = 6;
    COPY_CONFIGS = 7;
    LAG_SETTINGS = 8;
    IP_PORTS = 9;
    VE_PORTS = 10;
    PORT_CONFIGURATION = 11;
    OVERWRITE_CONFIGURATION = 12;
    STACK = 13;
    AAA_SETTING = 14;
    AAA_SERVER = 15;
}

message DeployLog {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    enum DeployStatus {
        PENDING = 1;
        STARTED = 2;
        SUCCESS = 3;
    FAILURE = 4;
    }

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string transactionId = 2;
    optional string groupId = 3;
    optional string modelFamily = 4;
    optional string yang = 5;
    optional ConfigType configType = 6;
    optional DeployScope deployScope = 7;
    optional DeployStatus deployStatus = 8;
    optional string switchIds = 9;
    optional int64 scheduled = 10;
    optional int64 startTime = 11;
    optional int64 endTime = 12;
    optional ConfigFeature configFeature = 13;
    optional int32 success = 14;
    optional int32 failed = 15;
    optional int32 failedNoResponse = 16;
    optional int32 failedSaveFlash = 17;
    optional string nodeId = 18;
}
```

```
message DeployLogItem {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    /**
     * @internal
     **/
    enum DispatchStatus {
        PENDING = 1;
        IN_PROGRESS = 2;
        SUCCESS = 3;
        FAILED = 4;
        PENDING_LOCAL_SYNC = 5;
        FAILED_NO_RESPONSE = 6;
        FAILED_SAVE_FLASH = 7;
        NO_CONFIG_CHANGE = 8;
        OFFLINE = 9;
        ROAMED = 10;
    }

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string transactionId = 2;
    optional string switchId = 3;
    optional string yang = 4;
    optional string clis = 5;
    optional DispatchStatus dispatchStatus = 6;
    optional string dispatchError = 7;
    optional int64 startTime = 8;
    optional int64 endTime = 9;
    optional string removeYang = 10;
    optional string nodeId = 11;
}

message SwitchConfigStore {
    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string yang = 2;
    optional string backup = 3;
    optional int64 lastConfigTime = 4;
    optional int64 lastSwitchConfigTime = 5;
    optional int64 lastStartupSwitchConfigTime = 6;
    optional bool localSyncCompleted = 7;
}

message CommonSettings {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional int64 updatedTime = 2;
    optional int64 createdTime = 3;
    optional string dnsConfig = 4;
}

message CommonSettingsDns {
    optional string ip = 1;
}

message StaticRoute {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string groupId = 2;
    optional string familyId = 3;
    optional string destinationIp = 4;
    optional string nextHop = 5;
    optional int64 adminDistance = 6;
    optional PushTimeType pushTimeType = 7;
    optional int64 pushTime = 8;

    optional int64 createdTime = 9;
    optional int64 updatedTime = 10;
```

```
    optional string switchId = 11;
    optional int64 scheduled = 12;
    optional bool initialConfig = 13;
}

message ACLConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string groupId = 2;
    optional string familyId = 3;
    optional string switchId = 4;
    optional string name = 5;
    optional ACLType aclType = 6;
    optional string ruleConfig = 7;

    optional PushTimeType pushTimeType = 8;
    optional int64 pushTime = 9;

    optional int64 createdTime = 10;
    optional int64 updatedTime = 11;
    optional int64 scheduled = 12;
}

enum ACLType {
    STANDARD = 1;
    EXTENDED = 2;
}

message ACLRule {
    optional int32 seq = 1;
    optional ACLRuleAction action = 2;
    optional ACLRuleProtocol protocol = 3;
    optional string srcNetwork = 4;
    optional string destNetwork = 5;
    optional int32 srcPort = 6;
    optional int32 destPort = 7;
}

enum ACLRuleAction {
    PERMIT = 1;
    DENY = 2;
}

enum ACLRuleProtocol {
    IP = 1;
    TCP = 2;
    UDP = 3;
}

enum PushTimeType {
    NOW = 1;
    SCHEDULE = 2;
}

message StackConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id             = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional int64  createdTime    = 2;
    optional int64  updatedTime    = 3;

    optional string switchId       = 4;
    optional bool   isActiveRole   = 5;
    optional string activeSwitchId = 6;
    optional int32  suggestedId    = 7;
    optional StackDeployState stackDeployState = 8;

    enum StackDeployState {
        INIT = 1;
        FAILURE = 2;
```

```
        SUCCESS = 3;
    }
}

message SpecificSettings {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string hostname = 2;
    optional IgmpSnoopingType igmpSnooping = 3;
    optional bool jumboMode = 4;
    optional bool dhcpServerEnabled = 5;
    optional bool enabledAclPerPortPerVlan = 6;
    optional string dhcpServerSetting = 7;

    optional int64 createdTime = 8;
    optional int64 updatedTime = 9;
    optional string groupId = 10;

    optional string ipv4DhcpSnoopingTrustPort = 11;
    optional string arpInspectionTrustPort = 12;
}

message DHCPServer {
    optional string poolName = 1;
    optional string network = 2;
    optional string excludedStart = 3;
    optional string excludedEnd = 4;
    optional int32 leaseDays = 5;
    optional int32 leaseHrs = 6;
    optional int32 leaseMins = 7;
    optional string defaultRouterIp = 8;
    repeated DHCPOption dhcpOptions = 9;
}

message DHCPOption {
    optional int32 seq = 1;
    optional DHCPOptionType type = 2;
    optional string value = 3;


    enum DHCPOptionType {
        ASCII = 1;
        HEX = 2;
        IP = 3;
    }
}

enum IgmpSnoopingType {
    IST_NONE = 1;
    ACTIVE = 2;
    PASSIVE = 3;
}

message PortConfiguration {
  option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

  optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
  optional string groupId = 2;
  optional string switchId = 3;
  optional string port = 4;
  optional string portName = 5;
  optional bool portEnabled = 6;
  optional string taggedVlans = 7;
  optional string untaggedVlans = 8;
  optional bool poeEnabled = 9;
  optional POEClass poeClass = 10;
  optional int32 poePriority = 11;
  optional PortSpeed portSpeed = 13;
  optional bool rstpAdminEdgePortEnabled = 14;
  optional bool stpBpduGuardEnabled = 15;
  optional bool stpRootGuardEnabled = 16;
```

```
  optional bool dhcpSnoopingTrustPortEnabled = 17;
  optional bool ipsgEnabled = 18;
  optional bool lldpEnabled = 19;
  optional string inAclConfigUUID = 20;
  optional string inAclConfigName = 21;
  optional string outAclConfigUUID = 22;
  optional string outAclConfigName = 23;
  optional string dhcpRelayAgent = 24;
  optional string ipAddress = 25;
  optional string ospfArea = 26;
  optional string subnetMask = 27;
  optional int64 createdTime = 28;
  optional int64 updatedTime = 29;
  optional string portIdentifierFormatted = 30;
  optional bool hasLayerThreeConfig = 31;
  optional bool poeCapability = 32;

  enum POEClass{
    ZERO = 1;
    ONE = 2;
    TWO = 3;
    THREE = 4;
    FOUR = 5;
    FIVE = 6;
  }

  enum PortSpeed {
    NONE = 1;
    AUTO = 2;
    TEN_M_FULL = 3;
    TEN_M_HALF = 4;
    ONE_HUNDRED_M_FULL = 5;
    ONE_HUNDRED_M_HALF = 6;
    ONE_G_FULL = 7;
    ONE_G_FULL_MASTER = 8;
    ONE_G_FULL_SLAVE = 9;
    TWO_POINT_FIVE_G_FULL = 10;
    TWO_POINT_FIVE_G_FULL_MASTER = 11;
    TWO_POINT_FIVE_G_FULL_SLAVE = 12;
    FIVE_G_FULL = 13;
    FIVE_G_FULL_MASTER = 14;
    FIVE_G_FULL_SLAVE = 15;
    TEN_G_FULL = 16;
    TEN_G_FULL_MASTER = 17;
    TEN_G_FULL_SLAVE = 18;
    TWENTY_FIVE_G_FULL = 19;
    FORTY_G_FULL = 20;
    ONE_HUNDRED_G_FULL = 21;
  }

}

enum SpanningTreeType {
    STT_NONE = 1;
    STP = 2;
    RSTP = 3;
}

message VEConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string name = 2;
    optional string groupId = 3;
    optional string switchId = 4;
    optional string ospfArea = 5;
    optional int32 vlanId = 6;
    optional string dhcpRelayAgent = 7;
    optional string ipAddress = 8;
    optional string subnetMask = 9;
    optional string inAclConfigUUID = 10;
    optional string inAclConfigName = 11;
    optional string outAclConfigUUID = 12;
```

```
    optional string outAclConfigName = 13;
    optional int32 veId = 14;
    optional int64 createdTime = 15;
    optional int64 updatedTime = 16;
    optional bool initialConfig = 17;
}

message LAGConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string switchId = 2;
    optional string groupId = 3;
    optional string name = 4;
    optional string portsConfig = 5;
    optional LAGType type = 6;
    optional int64 createdTime = 7;
    optional int64 updatedTime = 8;
    optional string lastName = 9;

    /**
     * @internal
     **/
    enum LAGType {
        STATIC = 1;
        DYNAMIC = 2;
    }
}

message VlanPortRelation {
  option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

  optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];

  optional string groupId = 2;
  optional string switchId = 3;
  optional RelationType relationType = 4;
  optional string port = 5;
  optional int32 vlanId = 6;

  /**
   * @internal
   **/
  enum RelationType{
    TAGGED = 1;
    UNTAGGED = 2;
  }
}

message DeploySchedule {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional int64 scheduled = 2;
    optional int64 createdTime = 3;
    optional int64 pickedTime = 4;
    optional string transactionId = 5;
    optional string entityId = 6;
    optional string groupId = 7;
    optional string switchId = 8;
    optional string modelFamily = 9;
    optional ConfigType configType = 10;
    optional DeployScope deployScope = 11;
    optional ConfigFeature configFeature = 12;
    optional string config = 13;
    optional string switchIds = 14;
    optional string works = 15;
    optional string nodeId = 16;
}

message DeployPending {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
```

```
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional int64 createdTime = 2;
    optional string switchId = 3;
    optional ConfigType configType = 4;
    optional ConfigFeature configFeature = 5;
    optional string globalTypes = 6;
    optional string config = 7;
    optional string nodeId = 8;
}

message SwitchClientStatus {
    /**
    * @property snapshot
    * @aggregation NULL
    * @description List of Switch Client Visibility
    * @since 5.1.2.1
    */
    repeated SwitchClientVisibility switchClientVisibilities = 1;
    /**
    * @property snapshot
    * @aggregation NULL
    * @description Number of Switch Client Visibility
    * @since 5.1.2.1
    */
    optional int32 switchClientVisibilityCount = 2;
    /**
    * @property snapshot
    * @aggregation NULL
    * @description Domain identifier of SZ.
    * @since 5.1.2.1
    */
    optional string domainId = 3;
    /**
     * @property snapshot
     * @aggregation NULL
     * @description Identifier of switch.
     * @since 5.1.2.1
     */
    optional string switchId = 4;
}

message SwitchClientVisibility {
    option (com.ruckuswireless.scg.protobuf.storage.category) = INFORMATION;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true]; //SwitchId +
SwitchPortId + ClientMAC + VLAN
    optional string domainId = 2;
    optional string tenantId = 3;
    optional string switchId = 4;
    optional string groupId = 5;
    optional string unitId = 6;
    optional string switchPortId = 7;
    optional string domainName = 8;
    optional string switchName = 9;
    optional string switchPort = 10;
    optional string clientMac = 11;
    optional string clientVlan = 12;
    optional ClientType clientType = 13;
    optional ClientAuthType clientAuthType = 14;
    optional ClientAuthStatus clientAuthStatus = 15;
    optional string clientDesc = 16;
    optional string clientUserName = 17;
    optional string clientIpv4Addr = 18;
    optional string clientIpv6Addr = 19;
    optional string clientUpTime = 20;
    optional string pastAuthHistory = 21;
    optional int64 createdTime = 22;
    optional int64 updatedTime = 23;
    optional string vlanName = 24;
    optional int64 historyExpirationTime = 25;
    optional string switchGroupLevelOneId = 26;
    optional string switchGroupLevelTwoId = 27;
    optional string switchGroupLevelOneName = 28;
```

```
    optional string switchGroupLevelTwoName = 29;
    optional bool isRuckusAP = 30;
    optional string clientName = 31;
    optional string dot1XIpv4Addr = 32;
    optional string dot1XIpv6Addr = 33;
    optional string clientVni = 34;
}

enum ClientType {
    OTHER = 1;
    PHONE = 2;
    WLAN_AP = 3;
    ROUTER = 4;
    BRIDGE = 5;
    CABLE_DEVICE = 6;
}

enum ClientAuthType {
    CLIENT_AUTH_TYPE_NONE = 1;
    DOT1X = 2;
    MAC_AUTH = 3;
    WEB_AUTH = 4;
}

enum ClientAuthStatus {
    NO_AUTH = 1;
    ALLOWED = 2;
    BLOCKED = 3;
    RESTRICTED = 4;
    CRITICAL = 5;
    GUEST = 6;
}

message SwitchConfigurationMessage {
    optional SwitchClusterMessage clusterInfo = 1;
    optional uint64 timestamp = 2;
}

message SwitchClusterMessage {
    optional string clusterUuid = 1;
    optional string clusterName = 2;
    repeated TenantMessage tenantInfos = 3;
}

message TenantMessage {
    optional string tenantId = 1;
    optional string tenantName = 2;
    optional DomainMessage adminDomain = 3;
}

message DomainMessage {
    optional string domainId = 1;
    optional string domainName = 2;
    repeated DomainMessage subDomainInfos = 3;
    repeated SwitchGroupMessage switchGroupInfos = 4;
}

message SwitchGroupMessage {
    optional string switchGroupId = 1;
    optional string switchGroupName = 2;
    repeated SwitchGroupMessage subSwitchGroupInfos = 3;
}

message RealtimeSwitchStatus {
    optional string serialNumber = 1;
    optional string switchMac = 2;
    optional string domainId = 3;
    optional string domainName = 4;
    optional string switchGroupLevelOneId = 5;
    optional string switchGroupLevelOneName = 6;
    optional string switchGroupLevelTwoId = 7;
    optional string switchGroupLevelTwoName = 8;
```

```
    optional string status = 9;
    optional bool operational = 10;
    optional bool firmwareUpgrading = 11;
    optional string ipAddress = 12;
    optional string subnetMask = 13;
    optional string defaultGateway = 14;
    optional string staticOrDynamic = 15;
    optional string dns = 16;
    optional string firmwareVersion = 17;
    optional bool collectorExecuted = 18;
}

message SupportSaveStatus {
    option (com.ruckuswireless.scg.protobuf.storage.category) = GROUP;
    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string serialNumber = 2;
    optional string switchId = 3;
    optional string scpIp = 4;
    optional int32 scpPort = 5;
    optional string userName = 6;
    optional string password = 7;
    optional string pathToSave = 8;
    optional DownloadStatus downloadStatus = 9;
    optional string createdTime = 10;

    enum DownloadStatus{
        DOWNLOADING = 1;
        DONE = 2;
        TIMEOUT = 3;
        FAILED = 4;
    }
}

message KumoSwitch {
  option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;

  optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
  optional string initialCli = 2;
  optional string portConfiguration = 3;
  optional string serialNumber = 4;
  optional string featureConfigs = 5;
}

message RelayExternalSyslogConfig {
    option (com.ruckuswireless.scg.protobuf.storage.category) = CONFIG;

    optional string id = 1 [(com.ruckuswireless.scg.protobuf.storage.id) = true];
    optional string ip = 2;
    optional int32 port = 3;
    optional string protocol = 4;
}

message SwitchDetail {
    optional string id = 1;
    optional string switchName = 2;
    optional string macAddress = 3;
    optional string model = 4;
    optional string family = 5;
    optional string ipAddress = 6;
    optional string registrationStatus = 7;
    optional int32 numOfPorts = 8;
    optional string serialNumber = 9;
    optional string groupName = 10;
    optional string groupId = 11;
    optional string status = 12;
    optional string firmwareVersion = 14;
    optional bool isStack = 15;
    optional uint64 numOfUnits = 16;
    optional string szNodeAttached = 17;
    optional bool operational = 18;
}
```

```
message SwitchDetailMessage {
    repeated SwitchDetail switchDetail = 1;
}

message unitSlNum {
    required string id = 1;
    optional string remotePortMac = 2;
    optional string remoteDeviceName = 3;
    optional string remotePortType = 4;
    optional string remotePortDesc = 5;
    optional string localPort = 6;
    optional string remotePort = 7;
    optional string isRuckusAP = 8;
    optional string domainId = 9;
    optional string tenantId = 10;
    optional string switchGroupLevelOneId = 11;
    optional string switchGroupLevelTwoId = 12;
    optional string switchId = 13;
    optional string unitId = 14;
    optional string localPortIfaceName = 15;
    optional string localPortMac = 16;
}
```

Field Description

TABLE 80 Power Supply Group Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of switch.
stackId	string	snapshot	NULL	Stack identifier of switch.
powerSlotNum	int32	snapshot	NULL	Power Supply Slot Number of switch.
powerSupplyType	string	snapshot	NULL	Power Supply Type of switch.
powerSupplyStatus	string	snapshot	NULL	Power Supply Status of switch.
serialNumber	string	snapshot	NULL	Serial Number of switch.

TABLE 81 Fan Group Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of switch.
stackId	string	snapshot	NULL	Stack identifier of switch.
slotNum	int32	snapshot	NULL	Fan Slot Number of switch.
type	string	snapshot	NULL	Fan Type of switch.
status	string	snapshot	NULL	Fan Status of switch.
serialNumber	string	snapshot	NULL	Serial Number of switch.

TABLE 82 Switch Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of switch.
firmware	string	snapshot	NULL	Firmware version of switch.

TABLE 82 Switch Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
model	string	snapshot	NULL	Model of switch.
defaultGateway	string	snapshot	NULL	Default gateway of switch.
numOfPorts	int32	snapshot	NULL	Number of switch ports.
natIp	string	snapshot	NULL	NAT IP of switch.
status	string	snapshot	NULL	Status of switch
poeAvailable	bool	snapshot	NULL	PoE support of switch.
lastBackup	uint64	snapshot	NULL	Last backup of switch.
cpu	uint64	snapshot	NULL	CPU utilization of switch.
memory	uint64	snapshot	NULL	Memory utilization of switch.
uptime	string	snapshot	NULL	Uptime of switch.
alerts	int32	snapshot	NULL	Alerts of switch.
isStack	bool	snapshot	NULL	Stack support of switch.
stackId	string	snapshot	NULL	Stack identifier of switch.
priority	int32	snapshot	NULL	Priority.
modules	string	snapshot	NULL	Modules of switch.
domainId	string	snapshot	NULL	Domain identifier of SZ.
groupId	string	snapshot	NULL	Group identifier of switch.
lastUpdateTimeInMillis	uint64	snapshot	NULL	Last update time in millis.
switchName	string	snapshot	NULL	Name of switch.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
serialNumber	string	snapshot	NULL	Serial number of switch.
unitSerialNumbers	string	snapshot	NULL	Serial numbers of switch unit.
portModuleIds	string	snapshot	NULL	Port module identifiers of switch.
partitionInUse	string	snapshot	NULL	Partition in use.
switchMode	string	snapshot	NULL	Mode of switch.
switchSWVersion	string	snapshot	NULL	Software version of switch.
numOfUnits	int32	snapshot	NULL	Number of switch units.
poeUtilization	int32	snapshot	NULL	PoE allocated capacity of switch.
poeTotal	int32	snapshot	NULL	Total PoE capacity of switch.
poeFree	int32	snapshot	NULL	PoE unallocated capacity of switch.
family	string	snapshot	NULL	Family of switch.
cloudPort	string	cloudPort	NULL	Cloud port(uplink port) of switch for Alto.
domainName	string	snapshot	NULL	Domain name of SZ.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch.
powerSupplyGroups	string	snapshot	NULL	Power Supply Group status of switch.
fanGroups	string	snapshot	NULL	Fan Group status of switch.

TABLE 82 Switch Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
ipAddress	string	snapshot	NULL	Switch IP.
subnetMask	string	snapshot	NULL	Switch subnetmask.
staticOrDynamic	string	snapshot	NULL	Switch default gateway.
dns	string	snapshot	NULL	Switch DNS settings.`PortStatus`
temperatureGroups	string	snapshot	NULL	Temperature Group status of switch.
totalMemory	uint64	snapshot	NULL	Total memory.
freeMemory	uint64	snapshot	NULL	Free memory.
hasPoECapability	BOOL	snapshot	NULL	Has PoE capability.
operational	BOOL	snapshot	NULL	operational
reservedField1				
reservedField2				
reservedField3				
reservedField4				
localsyncStatus	.com.ruckuswireless.scg.protobuf.icx.LocalSyncRunningStatus	snapshot	NULL	localsyncStatus.poePdClass.

TABLE 83 Switch Stats Information

Attribute Name	**ValueType (size)**	**Property(Snapshot/ Delta/Serialization)**	**ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)**	Description
id	string	snapshot	NULL	Switch identifier
timestamp	int64	snapshot	NULL	Timestamp.
switchSerialNum	string	snapshot	NULL	Serial number of switch.
tx	int64	snapshot	SUM	TX bytes of switch.
rx	int64	snapshot	SUM	RX bytes of switch.
txRx	int64	snapshot	SUM	TX/RX bytes of switch.
txPkt	int64	snapshot	SUM	TX packets of switch.
rxPkt	int64	snapshot	SUM	RX packets of switch.
txRxPkt	int64	snapshot	SUM	TX/RX packets of switch.
txRate	int64	snapshot	SUM	TX bit rate of switch (in kilobits per second, within a five-minute interval.).
rxRate	int64	snapshot	SUM	RX bit rate of switch (in kilobits per second, within a five-minute interval.).
txRxRate	int64	snapshot	SUM	TX/RX bit rate of switch (in kilobits per second, within a five-minute interval.).
cpu	int64	snapshot	NULL	CPU utilization of switch.
memory	int64	snapshot	NULL	Memory utilization of switch.
domainId	string	snapshot	NULL	Domain identifier of SZ.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.

TABLE 83 Switch Stats Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
domainName	string	snapshot	NULL	Domain name of SZ.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch group.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch group.
multicastOut	int64	snapshot	SUM	MulticastOut
multicastIn	int64	snapshot	SUM	MulticastIn
broadcastOut	int64	snapshot	SUM	BroadcastOut
broadcastIn	int64	snapshot	SUM	broadcastIn
crcErr	int64	snapshot	SUM	CrcErr
outErr	int64	snapshot	SUM	OutErr
inErr	int64	snapshot	SUM	InErr
unicastOut	int64	snapshot	SUM	UunicastOut
unicastIn	int64	snapshot	SUM	UnicastIn
collectorExecuted	bool	snapshot	NULL	Switch collector executed

TABLE 84 Port Stats Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
id	string	snapshot	NULL	Port identifier
timestamp	int64	snapshot	NULL	Timestamp.
switchSerialNum	string	snapshot	NULL	Serial number of switch.
portMac	string	snapshot	NULL	Mac address of port.
tx	int64	snapshot	SUM	TX bytes of port.
rx	int64	snapshot	SUM	RX bytes of port
txRx	int64	snapshot	SUM	TX/RX bytes of port.
txPkt	int64	snapshot	SUM	TX packets of port.
rxPkt	int64	snapshot	SUM	RX packets of port.
txRxPkt	int64	snapshot	SUM	TX/RX packets of port.
txRate	int64	snapshot	SUM	TX bit rate of port (in kilobits per second, within a five-minute interval.).
rxRate	int64	snapshot	SUM	RX bit rate of port (in kilobits per second, within a five-minute interval.).
txRxRate	int64	snapshot	SUM	TX/RX bit rate of port (in kilobits per second, within a five-minute interval.).
domainId	string	snapshot	NULL	Domain identifier of SZ.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.

TABLE 84 Port Stats Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
switchId	string	snapshot	NULL	Identifier of switch.
domainName	string	snapshot	NULL	Domain name of SZ.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch.
switchUnitId	string	snapshot	NULL	Identifier of switch unit.

TABLE 85 Port Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
portMac	string	snapshot	NULL	Mac address of port.
switchId	string	snapshot	NULL	Identifier of switch.
type	string	snapshot	NULL	Type of port.
name	string	snapshot	NULL	Name of port.
status	string	snapshot	NULL	Status of port.
adminStatus	string	snapshot	NULL	Admin status of port.
vlanIds	string	snapshot	NULL	VLAN identifier of port.
neighborName	string	snapshot	NULL	Neighbor name of port.
portSpeed	string	snapshot	NULL	Speed of port.
ruckusDevice	bool	snapshot	NULL	Ruckus devices support of port.
lldpEnabled	bool	snapshot	NULL	LLDP enabled flag of port.
lagName	string	snapshot	NULL	LAG name of port.
lagStatus	string	snapshot	NULL	LAG status of port.
spanningTreeStatus	string	snapshot	NULL	Spanning tree status of port.
poeUsed	int32	snapshot	NULL	Amount of inline power consumed by the port. Each unit is in milliwatts.
poeTotal	int32	snapshot	NULL	Adjusts the inline power wattage. Valid values are from 1000 through 15400(IEEE802_3AF)/ 30000(IEEE802_3AT). Each unit is in milliwatts.
poePercent	double	snapshot	AVG	Percentage of inline power consumed by the port.
domainId	string	snapshot	NULL	Domain identifier of SZ.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
tx	int64	snapshot	SUM	TX bytes of port.
rx	int64	snapshot	SUM	RX bytes of port.
signalIn	double	snapshot	AVG	Input network utilization in hundredths of a percent over a five-minute interval.

TABLE 85 Port Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
signalOut	double	snapshot	AVG	Output network utilization in hundredths of a percent over a five-minute interval.
crcErr	int64	snapshot	NULL	Stats CRC align errors of port.
inErr	int64	snapshot	NULL	Input errors of port.
outErr	int64	snapshot	NULL	Output errors of port.
opticsType	string	snapshot	NULL	Optics type of port.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
portSpeedCapacity	string	snapshot	NULL	Capacity of port speed.
switchUnitId	string	snapshot	NULL	Identifier of switch unit.
portIfaceName	string	snapshot	NULL	Interface name of port.
poeEnabled	bool	snapshot	NULL	PoE enabled flag of port.
usedInFormingStack	bool	snapshot	NULL	"Used in forming stack" flag of port.
portIdentifier	string	snapshot	NULL	Description of port.
unTaggedVlan	string	snapshot	NULL	Untagged VLAN of port.
isInWarningState	bool	snapshot	NULL	"Is in warning state" flag of port.
inDiscard	int64	snapshot	NULL	Input discards of port.
broadcastIn	int64	snapshot	SUM	Input broadcast packets of port.
broadcastOut	int64	snapshot	SUM	Output broadcast packets of port.
multicastIn	int64	snapshot	SUM	Input multicast packets of port.
multicastOut	int64	snapshot	SUM	Output multicast packets of port.
poeType	string	snapshot	NULL	PoE type of port.
portIdentifierFormatted	string	snapshot	NULL	Formatted of port identifier.
portId	string	snapshot	NULL	Port identifier.
domainName	string	snapshot	NULL	Domain name of SZ.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch.
unicastOut	INT64	snapshot	NULL	UnicastOut
unicastIn	INT64	snapshot	NULL	UnicastIn
vlanDetailInformation	string	snapshot	NULL	VLAN Detail Information
switchName	string	snapshot	NULL	switchName
neighborMacAddress	string	snapshot	NULL	Neighbor mac address of port.
inAclConfigName	string	snapshot	NULL	Ingress acl config name of port.
outAclConfigName	string	snapshot	NULL	egress acl config name of port.
poeClass	int32	snapshot	NULL	PoE class of port
poePriority	int32	snapshot	NULL	PoE priority of port
poeBudget	int32	snapshot	NULL	PoE budget of port
lagId	int32	snapshot	NULL	LAG ID
lagAdminStatus	string	snapshot	NULL	Admin status of LAG.

TABLE 85 Port Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
searchableVlans	string	snapshot	NULL	Searchable VLAN identifier of port
resourceName	PortType	snapshot	NULL	Identify the LAG / Physical port
isLagMember	bool	snapshot	NULL	The LAG port or Physical port
poePdClass	string	snapshot	NULL	A PD signature which the decive learns in the process of PD-classification.
poePdClassB	string	snapshot	NULL	Second PD class signature of dual signature PD which the device learns in the process of PD-classification.
poeLldpMaxPowerRequest	string	snapshot	NULL	Maximum power requested by PD through LLDP in milliwatts.
poeLldpMaxPowerRequestA	string	snapshot	NULL	In case of Dual Signature PD we have 2 channel, snAgentPoePortlldpReqPwrA is power in milliwatts for channel A.
poeLldpMaxPowerRequestB	string	snapshot	NULL	In case of Dual Signature PD we have 2 channel. snAgentPoePortlldpReqPwrB is power in milliwatts for channel B.
poePortCapability	.com.ruckuswireless.scg.protobuf.icx.PoePortCapability			Capability of port, 0-others, 1-2pair, 2-4pair.
poe2PairMaxPower	string	snapshot	NULL	Maximum power supported by 2pair port in milliwatts.
poe4PairMaxPower	string	snapshot	NULL	Maximum power supported by 4pair port in milliwatts.
poeOverdriveMode	.com.ruckuswireless.scg.protobuf.icx.PoeOverDriveMode	snapshot	NULL	Poe port overdrive mode. 0-none, 1-overdrive.

TABLE 86 Switch Unit Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of switch unit.
switchId	string	snapshot	NULL	Identifier of switch.
upTime	string	snapshot	NULL	Uptime of switch.
unitStatus	string	snapshot	NULL	Status of switch unit.
unitSlNum	string	snapshot	NULL	Serial Number of switch unit.
domainId	string	snapshot	NULL	Domain identifier of SZ.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch unit.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch unit.
domainName	string	snapshot	NULL	Domain name of switch unit.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch unit.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch unit.
unitId	int32	snapshot	NULL	stack unit ID.

TABLE 86 Switch Unit Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
unitState	string	snapshot	NULL	switch stacking config unit state.
unitName	string	snapshot	NULL	switch stacking config unit name.
poeUtilization	int32	snapshot	NULL	PoE allocated capacity of switch.
poeTotal	int32	snapshot	NULL	Total PoE capacity of switch.
poeFree	int32	snapshot	NULL	PoE unallocated capacity of switch.**SwitchUnitStatus**

TABLE 87 Connected Device Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN ,AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of remote port.
remotePortMac	string	snapshot	NULL	Mac address of remote port.
remoteDeviceName	string	snapshot	NULL	Name of remote device.
remotePortType	string	snapshot	NULL	Type of remote port.
remotePortDesc	string	snapshot	NULL	Description of remote port.
localPort	string	snapshot	NULL	Local port interface.
remotePort	string	snapshot	NULL	Remote port interface.
isRuckusAP	string	snapshot	NULL	RuckusAP support of remote device.
domainId	string	snapshot	NULL	Domain identifier of SZ.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
switchId	string	snapshot	NULL	Identifier of switch.
unitId	string	snapshot	NULL	Identifier of switch unit.
localPortIfaceName	string	snapshot	NULL	Interface name of local port.
localPortMac	string	snapshot	NULL	Mac address of local port.
domainName	string	snapshot	NULL	Domain name of SZ.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch.
updatedTime	int64	snapshot	NULL	Update time of connected device information
remoteDeviceMac	string	snapshot	NULL	ConnectedDevice remote mac address.

TABLE 88 LAG Status Information

Attribute Name	ValueType (size)	Property(Snapshot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,N ULL)	Attribute Name
lagName	string	snapshot	NULL	LAG port name.
lagId	int32	snapshot	NULL	LAG port ID.
lagStatus	string	snapshot	NULL	LAG port status.

TABLE 88 LAG Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Attribute Name
lagAdminStatus	string	snapshot	NULL	LAG port admin status.
vlanIds	string	snapshot	NULL	VLAN identifier of LAG port.
switchId	string	snapshot	NULL	Switch ID
unTaggedVlan	string	snapshot	NULL	Untagged VLAN of LAG port
switchUnitId	string	snapshot	NULL	Identifier of switch unit
domainName	string	snapshot	NULL	Domain name of SZ
switchGroupLevelOneName	string	snapshot	NULL	Level 1 identifier of switch group
resourceName	.com.ruckuswireless.scg.protobuf.icx.PortType	snapshot	NULL	Identify the LAG / Physical port
isLagMember	bool	snapshot	NULL	The LAG port or Physical port

TABLE 89 Switch Client Visibility Information

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
id	string	snapshot	NULL	Identifier of SwitchId + SwitchPortId + ClientMAC + VLAN.
domainId	string	snapshot	NULL	Domain identifier of SZ.
tenantId	string	snapshot	NULL	Tenant identifier of SZ.
switchId	string	snapshot	NULL	Identifier of switch.
groupId	string	snapshot	NULL	Group identifier of switch.
unitId	string	snapshot	NULL	Identifier of switch unit.
switchPortId	string	snapshot	NULL	Switch port ID.
domainName	string	snapshot	NULL	Domian name
switchName	string	snapshot	NULL	Switch name
switchPort	string	snapshot	NULL	Switch Port
clientMac	string	snapshot	NULL	Mac address of client.
clientVlan	string	snapshot	NULL	VLAN of client.
clientType	.com.ruckuswireless.scg.protobuf.icx.ClientType	snapshot	NULL	Type of client.
clientAuthType	.com.ruckuswireless.scg.protobuf.icx.ClientAuthType	snapshot	NULL	Auth type of client.
clientAuthStatus	.com.ruckuswireless.scg.protobuf.icx.ClientAuthStatus	snapshot	NULL	Status of client.
clientDesc	string	snapshot	NULL	Description of client.
clientUserName	string	snapshot	NULL	User name of client.
clientIpv4Addr	string	snapshot	NULL	IPv4 adder of client.
clientIpv6Addr	string	snapshot	NULL	IPv6 adder of client.

TABLE 89 Switch Client Visibility Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
clientUpTime	string	snapshot	NULL	Up time of client.
pastAuthHistory	string	snapshot	NULL	Past 24hour auth history of client.
createdTime	INT64	snapshot	NULL	Create time of client information.
updatedTime	INT64	snapshot	NULL	Update time of client information.
vlanName	string	snapshot	NULL	Name of Vlan.
historyExpirationTime	INT64	snapshot	NULL	Used to check if pastAuthHistory is expired.
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 Group name of switch.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 Group name of switch.
isRuckusAP	bool	snapshot	NULL	The wired client is RuckusAP
clientName	string	snapshot	NULL	Name of client
dot1XIpv4Addr	string	snapshot	NULL	Dot1x IPv4 address of client
dot1XIpv6Addr	string	snapshot	NULL	Dot1x IPv6 address of client
clientVni	string	snapshot	NULL	Client VNI

TABLE 90 Switch Configuration Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
clusterInfo	.com.ruckuswireless.scg.protobuf.icx.SwitchClusterMessage	snapshot	NULL	The cluster information.
timestamp	UINT64	snapshot	NULL	The timestamp

TABLE 91 Switch Cluster Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
clusterUuid	string	snapshot	NULL	Cluster UUID
clusterName	string	snapshot	NULL	Cluster name
tenantInfos	.com.ruckuswireless.scg.protobuf.icx.TenantMessage	snapshot	NULL	Group Tree System Hierarchy.

TABLE 92 Tenant Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
tenantId	string	snapshot	NULL	Tenant UUID
tenantName	string	snapshot	NULL	Tenant name

TABLE 92 Tenant Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
adminDomain	.com.ruckuswireless.scg.protobuf.icx.DomainMessage	snapshot	NULL	The admin domain of tenant.

TABLE 93 Domain Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
domainId	string	snapshot	NULL	Domain UUID
domainName	string	snapshot	NULL	Domain name
subDomainInfos	.com.ruckuswireless.scg.protobuf.icx.DomainMessage	snapshot	NULL	The list of sub-domains.
switchGroupInfos	.com.ruckuswireless.scg.protobuf.icx.SwitchGroupMessage	snapshot	NULL	The list of switch groups.

TABLE 94 Switch Group Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
switchGroupId	string	snapshot	NULL	Switch group UUID
switchGroupName	string	snapshot	NULL	Switch group name
subSwitchGroupInfos	.com.ruckuswireless.scg.protobuf.icx.SwitchGroupMessage	snapshot	NULL	The list of switch groups.

TABLE 95 Real time Switch Status Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
serialNumber	string	snapshot	NULL	Serial number of switch.
switchMac	string	snapshot	NULL	Mac address of switch.
domainId	int32	snapshot	NULL	Domain UUID
domainName	string	snapshot	NULL	Domain name
switchGroupLevelOneId	string	snapshot	NULL	Level 1 identifier of switch group.
switchGroupLevelOneName	string	snapshot	NULL	Level 1 group name of switch.
switchGroupLevelTwoId	string	snapshot	NULL	Level 2 identifier of switch group.
switchGroupLevelTwoName	string	snapshot	NULL	Level 2 group name of switch.
status	string	snapshot	NULL	Connection status of switch.
operational	bool	snapshot	NULL	Switch is operational
firmwareUpgrading	bool	snapshot	NULL	Switch firmware is upgrading
ipAddress	string	snapshot	NULL	Switch IP address.
subnetMask	string	snapshot	NULL	Switch subnetmask.

TABLE 95 Real time Switch Status Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
defaultGateway		snapshot	NULL	Switch default gateway.
staticOrDynamic		snapshot	NULL	Switch use static IP or DHCP.
dns		snapshot	NULL	Switch DNS settings.
firmwareVersion		snapshot	NULL	Switch firmware version.
collectorExecuted	bool	snapshot	NULL	Switch collector executed.

TABLE 96 Switch Detail Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
id	string	snapshot	NULL	Switch identifier
switchName	string	snapshot	NULL	Hostname of switch.
macAddress	string	snapshot	NULL	MAC address of switch
model	string	snapshot	NULL	Model of switch.
family	string	snapshot	NULL	Family of switch.
ipAddress	string	snapshot	NULL	Switch IP
registrationStatus	string	snapshot	NULL	Registration status of switch
numOfPorts	int32	snapshot	NULL	Number of switch ports.
serialNumber	string	snapshot	NULL	Serial number of switch
groupName	string	snapshot	NULL	Switch group name
groupId	string	snapshot	NULL	Switch group id
status	string	snapshot	NULL	Connection status of switch
firmwareVersion	string	snapshot	NULL	The firmware version of switch
isStack	bool	snapshot	NULL	The stack status
numOfUnits	uint64	snapshot	NULL	Number of units in the stack
szNodeAttached	string	snapshot	NULL	The SZ node id of the switch.
operational	bool	snapshot	NULL	Switch is operational

TABLE 97 Switch Detail Message Information

Attribute Name	ValueType (size)	Property(Snapshot/ Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN, AVG,NULL)	Description
switchDetail	SwitchDetail	snapshot	NULL	The list of attached switch detail

350 West Java Dr., Sunnyvale, CA 94089 USA
https://www.commscope.com