

Ruckus SmartZone GPB/MQTT Interface Getting Started Guide

Supporting SmartZone 3.6.2

Copyright, Trademark and Proprietary Rights Information

© 2018 ARRIS Enterprises LLC. All rights reserved.

No part of this content may be reproduced in any form or by any means or used to make any derivative work (such as translation, transformation, or adaptation) without written permission from ARRIS International plc and/or its affiliates ("ARRIS"). ARRIS reserves the right to revise or change this content from time to time without obligation on the part of ARRIS to provide notification of such revision or change.

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, ARRIS 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. ARRIS 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. ARRIS 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 ARRIS 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 ARRIS, ARRIS 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 ARRIS 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

ARRIS, the ARRIS logo, Ruckus, Ruckus Wireless, Ruckus Networks, Ruckus logo, the Big Dog design, BeamFlex, ChannelFly, EdgIron, FastIron, HyperEdge, ICX, IronPoint, OPENG, SmartCell, Unleashed, Xclaim, ZoneFlex are trademarks of ARRIS International plc and/or its affiliates. Wi-Fi Alliance, Wi-Fi, the Wi-Fi logo, the Wi-Fi CERTIFIED logo, Wi-Fi Protected Access (WPA), the Wi-Fi Protected Setup logo, and WMM are registered trademarks of Wi-Fi Alliance. Wi-Fi Protected Setup™, Wi-Fi Multimedia™, and WPA2™ are trademarks of Wi-Fi Alliance. All other trademarks are the property of their respective owners.

Contents

Preface.....	5
Document Conventions.....	5
Notes, Cautions, and Warnings.....	5
Command Syntax Conventions.....	6
Document Feedback.....	6
Ruckus Product Documentation Resources.....	6
Online Training Resources.....	7
Contacting Ruckus Customer Services and Support.....	7
What Support Do I Need?.....	7
Open a Case.....	7
Self-Service Resources.....	7
GPB-MQTT Interface Implementation.....	9
GPB-MQTT Overview.....	9
Prerequisite Task.....	9
Working with the GPB/MQTT Interface.....	11
Enabling Authentication in the MQTT Broker.....	11
Compiling Google Protobuf Binding Classes.....	14
Executing the Test Subscriber.....	15
Execution Script to Start Mosquitto MQTT	15
Execution Script Before Installing Release 5.0	15
Execution Script After Installing Release 5.0	16
Exit from the Test subscriber	16
Execution Result.....	16
Appendix.....	17
AP Message Hierarchy and Information.....	17
ap_avc.proto.....	19
ap_client.proto.....	22
ap_mesh.proto.....	26
ap_report.proto.....	29
ap_rogue.proto.....	43
ap_status.proto.....	45
ap_wired_client.proto.....	61
sci_event.proto.....	63
sci configuration message.....	65

Preface

- Document Conventions..... 5
- Command Syntax Conventions..... 6
- Document Feedback..... 6
- Ruckus Product Documentation Resources..... 6
- Online Training Resources..... 7
- Contacting Ruckus Customer Services and Support..... 7

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	<code>device(config)# interface ethernet 1/1/6</code>
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 .
<i>italics</i>	Publication titles	Refer to the <i>Ruckus Small Cell Release Notes</i> for more information.

Notes, Cautions, and 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.
<i>italic text</i>	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, <i>member[member...]</i> .
\	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.

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@arris.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.ruckuswireless.com>.

Online Training Resources

To access a variety of online Ruckus training modules, including free introductory courses to wireless networking essentials, site surveys, and Ruckus products, visit the Ruckus Training Portal at <https://training.ruckuswireless.com>.

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.ruckuswireless.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>

Preface

Contacting Ruckus Customer Services and Support

- Community Forums—<https://forums.ruckuswireless.com/ruckuswireless/categories>
- 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.

GPB-MQTT Interface Implementation

- GPB-MQTT Overview..... 9
- Prerequisite Task..... 9
- Working with the GPB/MQTT Interface..... 11
- Enabling Authentication in the MQTT Broker..... 11
- Compiling Google Protobuf Binding Classes..... 14
- Executing the Test Subscriber..... 15

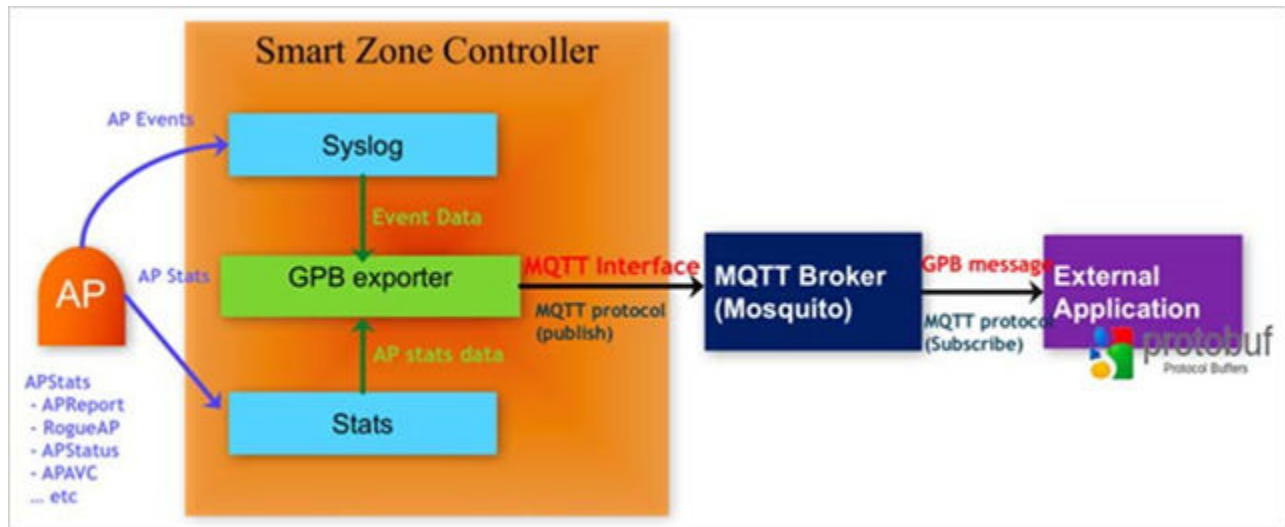
GPB-MQTT Overview

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

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

MQTT stands for MQ Telemetry Transport. It is a publish/subscribe, extremely simple and lightweight messaging protocol. It was designed as an extremely lightweight publish/subscribe messaging transport. It is useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium.

FIGURE 1 GPB/MQTT Interface Diagram



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, please refer to Oracle for downloading the latest JDK 8 from <http://www.oracle.com/technetwork/java/javase/overview/index.html>

3. Download and install gradle version 2.9.x.
Download from here: <https://gradle.org/>
4. Download and install Mosquitto 1.4.x broker and client and requisite libraries.
Download from here: <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//
repdata/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 mosquito mosquito-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 SZ with Mosquitto IP and port for it to transfer SCI data.
Refer to the SZ technical documentation to achieve this.
10. Enable SCI in the MQTT broker.
Refer to the "Working with the GPB/MQTT Interface" task.

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 on the SmartZone (SZ) 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.

Enabling Authentication in the MQTT Broker

The current implementation of SmartZone (SZ) needs authentication to the MQTT Broker.

Perform the following steps to create a 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 startup Mosquitto MQTT broker by user root or with sudo, it will result in an incorrect startup environment or other error.

1. Startup MQTT broker with security enabled.
 - a) After you install the mosquitto MQTT broker, you can start it up by the following command:

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

- b) 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
```

- c) According to the pskfile configuration of mosquitto.conf you need another pskfile to store the authentication pre-shared key. Here is another example for the pskfile:

```
testuser:7465737475736572
```

The value before the colon, “:,” sign represents the username which in this example is testuser. The value after the colon refers to the password phrase which is converted to hex representation. The original password text before converting is testuser.

The password has to match with the setting configured on controller SZ web interface.

NOTE

You can terminate the MQTT broker by pressing Ctrl + C on the MQTT broker console.

2. After starting one MQTT subscriber, you can start another MQTT subscriber to connect to the MQTT broker via an un-authenticated port 1883 or authenticated port 8883.

- a) Here is an example to connect to the MQTT broker with un-authenticated version:

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

- b) Here is an example to connect to the MQTT broker via 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 un-authenticated and authenticated version of the mosquitto_sub function, where the psk-identity and psk attribute is connected to MQTT broker via port number 8883.

3. Use the content in the following steps 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 SZs. 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. See the following figure.

FIGURE 2 Setting SCI Profile - Authenticated

The screenshot shows a dialog box titled "Create New SCI Profile" with a close button (X) in the top right corner. The dialog contains six input fields, each with a red asterisk indicating it is required:

- Name:
- Server Host:
- Server Port:
- User:
- Password:
- System ID:

At the bottom of the dialog are two buttons: "OK" and "Cancel".

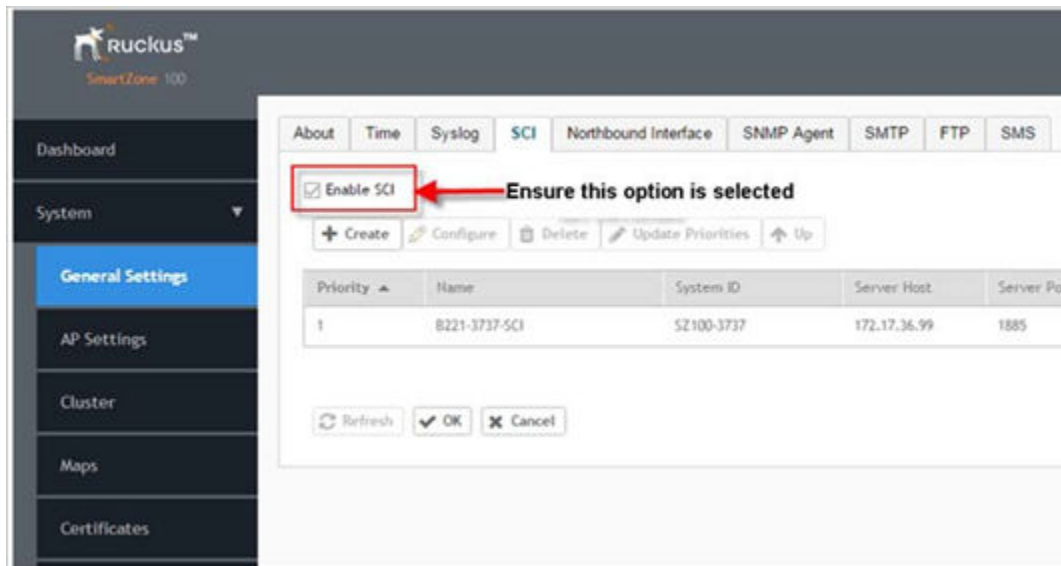
The SCI profile setting is used for allowing SZ to get the MQTT connection information. For example, IP address, port, user credential and so on. It does not mean that you must have a SCI setup to receive the GPB streaming data.

4. Save the SCI profile.

5. On the controller web user interface ensure the option **Enable SCI** is enabled. Navigate to **System > General Settings > SCI** on the controller web user interface as seen in Figure 2. Add the IP address for creating the MQTT channel to stream the data.

Ensure that you create the entry in the table to point the IP address to the installed MQTT server. For more details, refer to the *SmartZone Administration Guide*.

FIGURE 3 Enabling SCI 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 SmartZone (SZ) GPB .proto files from the Ruckus support site at: <https://support.ruckuswireless.com/contact-us>.
2. Follow the compiling instructions for getting the binding classes for different language. For more information, refer to <https://developers.google.com/protocol-buffers/>.

We can use ap_client.proto as an example for this task. The following steps will refer to the ap_client.proto as the example; you can substitute any .proto file.

3. If you have not installed the compiler, [download the package](#) and follow the instructions in the README file.
4. Run the compiler, specifying the source directory where your application's source code lives (the current directory is used if you do not provide a value), the destination directory where you want the generated code to go (usually the same as SRC_DIR), and the path to your .proto.

In this example, use ap_client.proto.

5. Now that you have `ap_client.proto` file, the next step is to 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 will be used for receiving SZ'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 SmartZone (SZ).

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 SZ GPB/MQTT interface.

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 Before Installing Release 5.0

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 which shows the statistical counter result.
- `n`— Normal mode which 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
```

Download the subscriber software `MockSCI-TLS-5.0.5.tar.gz`.

Execution Script After Installing Release 5.0

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 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 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
```

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, SmartZone) will be saved to the SciTlsMessages folder.

Appendix

- AP Message Hierarchy and Information..... 17
- ap_avc.proto..... 19
- ap_client.proto..... 22
- ap_mesh.proto..... 26
- ap_report.proto..... 29
- ap_rogue.proto..... 43
- ap_status.proto..... 45
- ap_wired_client.proto..... 61
- sci_event.proto..... 63
- sci configuration message..... 65

AP Message Hierarchy and Information

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

FIGURE 4 GPB Message Hierarchy

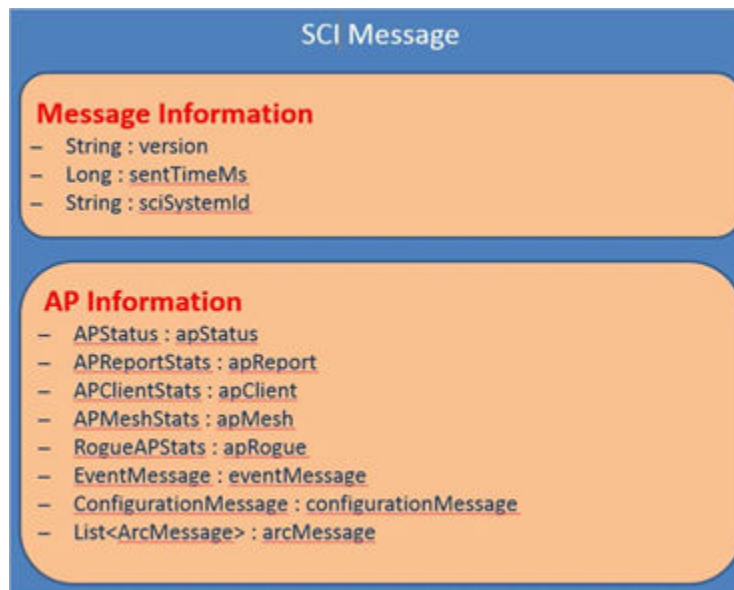
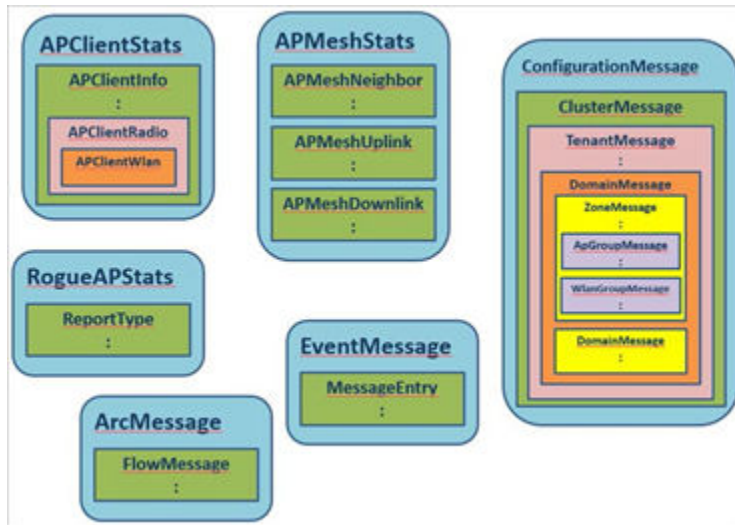


FIGURE 5 AP Information



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;
}

message UrlFilteringMsg {
  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;
}

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;
}

```

Field Description

TABLE 2 Flow message descriptions

Attribute Name	ValueType (size)	Property(Snapsh ot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,M IN,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	NULL	Rx bytes for this application
downlink	uint64	snapshot	NULL	Tx bytes for this application
total	uint64	snapshot	NULL	Tx + Rx bytes for the application
category	string	snapshot	NULL	Application category
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

TABLE 3 UrlFilteringMsg descriptions

Attribute Name	ValueType (size)	Property(Snapsh ot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,M IN,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

TABLE 4 Arc message descriptions

Attribute Name	ValueType (size)	Property(Snapsh ot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,M IN,AVG,NULL)	Description
version	string	snapshot	NULL	Arc dataversionnumber
zoneUUID	string	snapshot	NULL	Zone UUID
apgroupID	string	snapshot	NULL	AP group UUID
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	string	snapshot	NULL	Zone name

TABLE 4 Arc message descriptions (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
sampleTime	uint64	snapshot	NULL	AVC datasamplingtimestamp
aggregationInterval	uint32	snapshot	NULL	Stats aggregationinterval
apMac	string	snapshot	NULL	AP MAC address

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;
  /* 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;
}

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 dataplane_name = 18;
}

```

Field Description

TABLE 5 AP Client information

Attribute Name	ValueType (size)	Property(Snapsh ot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,M IN,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
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.

TABLE 5 AP Client information (continued)

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

TABLE 6 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
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	WLAN Group name
wlantenant_name	string	snapshot	NULL	WLAN tenant name

TABLE 7 AP Client Radio

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
mode	string	snapshot	NULL	Radio mode used by the AP's wireless interface
80211RadioMode	string	snapshot	NULL	Radio mode used by the AP
channel	int	snapshot	NULL	Current radio channel used by the AP
channelWidth	uint32	snapshot	NULL	Channel width used by the AP's radio
wlan	Struct APClientWlan	snapshot	NULL	WLAN information for this client

TABLE 8 AP Client Status

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

TABLE 8 AP Client Status (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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

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;
}

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;
}

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;
}
```

Field Description

TABLE 9 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	uint8	snapshot	NULL	Link status of the mesh downlink AP
rssI	char	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

TABLE 10 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	uint8	snapshot	NULL	Link status of the mesh uplink AP
rssI	char	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

TABLE 11 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	char	snapshot	NULL	RSSI of the neighboring AP

TABLE 12 AP Statistics 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
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

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 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;
}

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;
}

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 {
    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;
}

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;
}

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 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;
    }
}

```

Field Description

TABLE 13 AP Report Bin WLAN Information

Attribute Name	ValueType (size)	Property(Snapsh ot/Delta/ Serialization)	ValueAggregation Type(SUM,MAX,M IN,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

TABLE 13 AP Report Bin WLAN Information (continued)

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
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
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

TABLE 13 AP Report Bin WLAN Information (continued)

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

TABLE 14 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

TABLE 14 AP Report Bin Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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
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)

TABLE 14 AP Report Bin Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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)

TABLE 15 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.

TABLE 15 AP Report Bin Tunnel Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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 16 Hccid 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 17 Hccid 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
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
snr	float	snapshot	NULL	Client SNR (not implemented)
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 18 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

TABLE 19 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
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)...
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

TABLE 19 AP Report Bin Client Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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
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_estimate For example AP has client's throughput estimation of 10Mbps, 9Mbps, 9Mbps, 10Mbps, 0, 0, 0, 0, 0 . Then AP come out $(10+9+9+10)/4 = 9.5$ Mbps 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 10:09:30 in Bin(10:00~10:15), and the time is 10:15:30 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

TABLE 19 AP Report Bin Client Information (continued)

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

TABLE 20 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

TABLE 20 AP Report Bin Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
				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 21 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	snapshot	NULL	Total number of bin data.
binClient	Struct APReportBinClient	snapshot	NULL	Total number of Client bin data.
binIPSec	Struct APReportBinIPSec	snapshot	NULL	The number of IPSec tunnel.
binTunnel	Struct APReportBinTunnel	snapshot	NULL	The number of tunnel data.
binRadio	Struct APReportBinRadio	snapshot	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

TABLE 21 AP Report Stats Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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

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;
}

```

Field Description

TABLE 22 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.

TABLE 22 AP Rogue Report Type Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot /Delta/ Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
rsssi	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 23 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

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;
}

message APStatusIPSecStats {
  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;
}

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;
/* 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 DisassocAbnormal = 1028;
optional uint32 NumDisassocCapacity = 1029;
optional uint32 NumDisassocLeave = 1030;
optional uint32 NumDisassocMisc = 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;
}

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 wlangs = 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;
/* 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;
}

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 dnatsInfo = 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;
  /* 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;
}

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 cmStatusRangingAborted = 22;
  optional uint32 cmStatusDcslsOperMode = 23;
}

```

```

        optional uint32 cmStatusModulationType = 24;
        optional string cmStatusEqualizationData =25;
    }

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

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;
}

```

Field Description

TABLE 24 AP Status 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
type	int	snapshot	NULL	Tunnel type
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

TABLE 24 AP Status Tunnel Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot /Delta/ Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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
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

TABLE 25 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

TABLE 26 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 proprietary 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 proprietary internal design.
rxBcastFrames	uint64	snapshot	SUM	Total broadcast packets received
rxMcastFrames	uint64	snapshot	SUM	Total multicast packets received

TABLE 26 AP Status WLAN Information (continued)

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

TABLE 27 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

TABLE 27 AP Status Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
				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
ap	string	snapshot	NULL	AP mac address
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

TABLE 27 AP Status Radio Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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

TABLE 28 AP Status Brownout Information

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

TABLE 29 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)

TABLE 29 AP Status IPsec Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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 30 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
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 could configure 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

TABLE 30 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
				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)
ip	string	snapshot	NULL	IPv4 address of the AP
ipv6	string	snapshot	NULL	IPv6 address of the AP
ipseclp	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

TABLE 30 AP Status System Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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, poe, auto)

TABLE 31 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)
sfplInfo	string	snapshot	NULL	Sfp supported information

TABLE 32 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
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.

TABLE 32 Cable Modem Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
cmStatusCode	string	snapshot	NULL	Gets the data for status code.
cmStatusTxPower	string	snapshot	NULL	Gets the data for <i>TxPower</i> .
cmStatusResets	uint32	snapshot	NULL	Gets the data for status reset.
cmStatusLostSyncs	uint32	snapshot	NULL	Gets the data for <i>StatusLostsSyncs</i> .
cmStatusInvalidMaps	uint32	snapshot	NULL	Gets the data <i>StatusInvalidMap</i> .
cmStatusInvalidUcdfs	uint32	snapshot	NULL	Gets the data <i>StatusInvalidUcdfs</i> .
cmStatusInvalidRangingResponses	uint32	snapshot	NULL	Gets the data <i>StatusInvalidRangingResponses</i> .
cmStatusInvalidRegistrationResponses	uint32	snapshot	NULL	Gets the data for <i>StatusInvalidRegistrationResponses</i>
cmStatusT1Timeouts	uint32	snapshot	NULL	Gets the data for <i>StatusT1Timeouts</i> .
cmStatusT2Timeouts	uint32	snapshot	NULL	Gets the data <i>StatusT2Timeouts</i> .
cmStatusT3Timeouts	uint32	snapshot	NULL	Gets the data for <i>StatusT3Timeouts</i> .
cmStatusT4Timeouts	uint32	snapshot	NULL	Gets the data for <i>StatusT4Timeouts</i> .
cmStatusRangingAbortedds	uint32	snapshot	NULL	Gets the data for <i>statusRangingAbortedds</i>
cmStatusDocsisOperMode	uint32	snapshot	NULL	Gets the data for <i>StatusDocsisOperMode</i> .
cmStatusModulationType	uint32	snapshot	NULL	Gets the data for <i>CmStatusModulationType</i> .
cmStatusEqualizationData	string	snapshot	NULL	Gets the data for <i>CmStatusEqualizationData</i> .

TABLE 33 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 34 AP Status Data Information

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
APSystem	struct	NULL	NULL	Structure name
APIPSec	struct	NULL	NULL	Structure name
APBrownout	struct	NULL	NULL	Structure name
APRadio	struct	NULL	NULL	Structure name
APIPSecStats	struct	NULL	NULL	Structure name

TABLE 34 AP Status Data Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
APTunnel	struct	NULL	NULL	Structure name
IanPortStatus	struct	NULL	NULL	Structure name
certificateReload	struct	NULL	NULL	Structure name
cableModemInfo	struct	NULL	NULL	Structure name
APStatusLBS	struct	NULL	NULL	Structure name

TABLE 35 AP Status Data

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
ap_status_data	struct	NULL	NULL	Structure name
zone_id	string	snapshot	NULL	Zone UUID. Unique zone ID (for example, b381206b-2e5d-43dc-b249-e36ffae9855c) assigned by the controller.
apgroup_id	string	snapshot	NULL	AP Group identifier
cluster_id	string	snapshot	NULL	Cluster identifier
domain_id	string	snapshot	NULL	Domain UUID
aptenant_id	string	snapshot	NULL	AP tenant UUID
map_id	string	snapshot	NULL	SZ MAP UUID
aptenant_name	string	snapshot	NULL	AP tenant name
zone_name	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. The AP retains the zone name.
apgroup_name	string	snapshot	NULL	AP Group name.
domain_name	string	snapshot	NULL	Domain name.
wlangroup24G_id	string	snapshot	NULL	Wlan group for 2.4G UUID
wlangroup24G_name	string	snapshot	NULL	Wlan group for 2.4G name
wlangroup5G_id	string	snapshot	NULL	Wlan group for 5G UUID
wlangroup5G_name	string	snapshot	NULL	Wlan group for 5G name
sampleTime	uint64	snapshot	NULL	Sampling time for this report
aggregationInterval	uint32	snapshot	NULL	Stats aggregating interval
map_name	string	snapshot	NULL	SZ MAP name
apMac	string	snapshot	NULL	Access Point MAC address

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;
}

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;
}

```

Field Description

TABLE 36 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
timestamp	uint64	Snapshot	NULL	time for this report
sampleTime	uint64	Snapshot	NULL	sample time for these stats

TABLE 36 AP Wired Client Stats Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
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

TABLE 37 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

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;
}
message MessageEntry {
  optional string key = 1;
  optional string value = 2;
}

```

Field Description

TABLE 38 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
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

TABLE 38 Event Message Information (continued)

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
clientMac	string	snapshot	NULL	UE/Client MAC address if the events are related to the client
timestamp	uint64	snapshot	NULL	Event timestamp (UTC time)
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

TABLE 39 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 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 40 Message Information

Attribute Name	ValueType (size)	Property(Snapshot/Delta/Serialization)	ValueAggregation Type(SUM,MAX,MIN,AVG,NULL)	Description
version	uint32	snapshot	NULL	GPB version
timestamp	uint64			

TABLE 41 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

TABLE 42 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 43 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

TABLE 44 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

TABLE 45 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 46 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
version	uint32	snapshot	NULL	GPB version
timestamp	uint64			

TABLE 47 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



© 2018 ARRIS Enterprises LLC. All rights reserved.
Ruckus Wireless, Inc., a wholly owned subsidiary of ARRIS International plc.
350 West Java Dr., Sunnyvale, CA 94089 USA
www.ruckuswireless.com