SMTP client Application

User guide

Version 0.2

May 2016

Redpine Signals, Inc.
2107 N. First Street, #540
San Jose, CA 95131.
Tel: (408) 748-3385
Fax: (408) 705-2019
Email: info@redpinesignals.com
Website: www.redpinesignals.com

About this Document

This document describes the process of bringing up the RS9113 based module as a SMTP client.

Disclaimer:

The information in this document pertains to information related to Redpine Signals, Inc. products. This information is provided as a service to our customers, and may be used for information purposes only. Redpine assumes no liabilities or responsibilities for errors or omissions in this document. This document may be changed at any time at Redpine's sole discretion without any prior notice to anyone. Redpine is not committed to updating this document in the future.

Table of Contents

Table of Figures

Table of Tables

No table of figures entries found.

1 Introduction

This project is applicable to all the WiSeConnect variants like WiSeConnect Plus, WiSeMCU and WyzBee. The term WiSeConnect refers to its appropriate variant.

1.1 Protocol Ovserview

SMTP is a Simple Mail Transfer Protocol, which is used to send mails from mail client to mail server. SMTP is a connection-oriented, text-based protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically a Transmission Control Protocol (TCP) connection.

An SMTP session consists of commands originated by an SMTP client (the initiating agent, sender, or transmitter) and corresponding responses from the SMTP server (the listening agent, or receiver) so that the session is opened, and session parameters are exchanged

1.2 Application Overview

1.2.1 Overview

This application demonstrates how WiSeConnect device sends a mail to SMTP server.

In this application, WiSeConnect device connects to Access Point in station mode and connects to SMTP server using SMTP client. After successful connection with SMTP server, application sends a mail to SMTP server.

1.2.2 Sequence of Events

This Application explains user how to:

- Connect to Access Point
- Connect with SMTP server opened in remote peer
- Send a mail to remote SMTP server

1.3 Application Setup

The WiSeConnect in its many variants supports SPI and UART interfaces. Depending on the interface used, the required set up is as below:

1.3.1 SPI based Setup Requirements

- Windows PC1 with CooCox IDE
- Spansion (MB9BF568NBGL) micro controller

Note: If user does not have Spansion (MB9BF568NBGL) host platform, please go through the SPI-Porting guide **\sapis\docs\RS9113-WiSeConnect-SAPI-Porting-Guide-vx.x.pdf** for SAPIs porting to that particular platform.

- WiSeConnect device
- WiFi Access point
- Windows PC2 with a simple SMTP server installed.

Note: Download simple SMTP server from the below link,
http://nilhcem.github.com/FakeSMTP/downloads/fakeSMTP-latest.zip

1.3.2 UART/USB-CDC based Setup Requirements

- Windows PC1 with Dev-C++ IDE
- WiSeConnect device
- WiFi Access point
- Windows PC2 with a simple SMTP server installed.

Note: Download simple SMTP server from the below link,
http://nilhcem.github.com/FakeSMTP/downloads/fakeSMTP-latest.zip

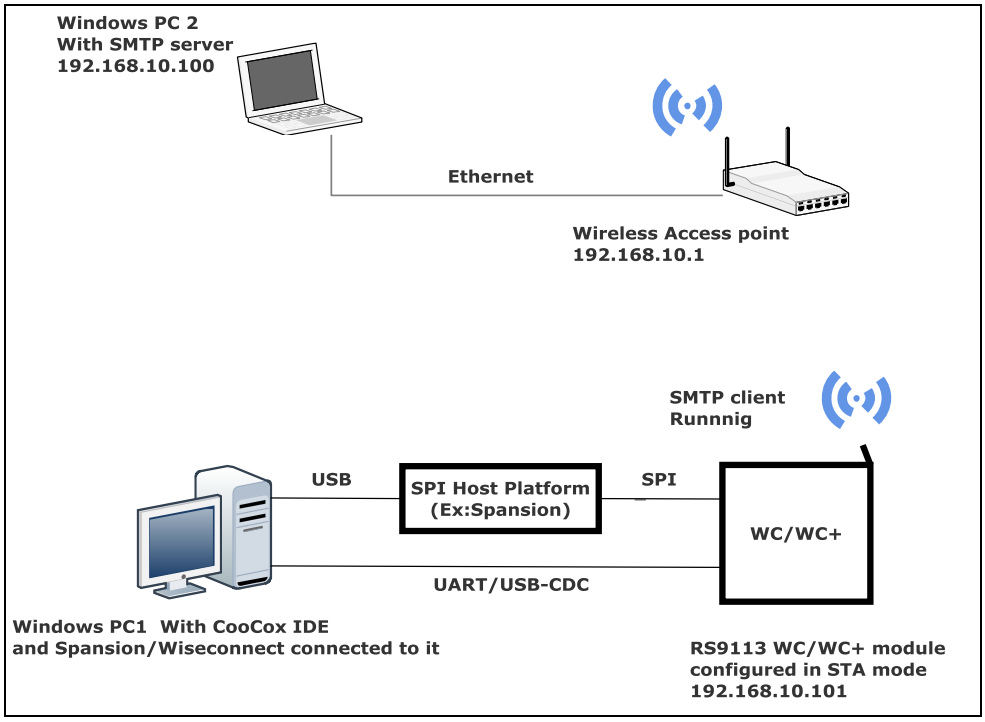


Figure 1: Setup Diagram

2 Configuration and Execution of the Application

The example application is available in the Release at **{Release $}/host/sapis/examples.** These examples will have to be initialized, configured and executed to test the application. The initialization varies based on the interface but configuration and execution are the common.

2.1 Initializing the Application

2.1.1 SPI Interface

If User using SPI interface, Please refer the document ***sapis/platforms/spansion_MB9BF568NBGL/RS9113-WiSeConnect_SAPIS_Spansion_Project_User_guide.pdf*** for opening the ***smtp_client*** example in CooCox IDE.

2.1.2 UART/USB-CDC Interface

If User using UART interface, Please refer the document ***sapis/platforms/windows_uart/RS9113-WiSeConnect_SAPIS_Windows_Project_UserGuide.pdf*** for opening the ***smtp_client*** example in Dev-C++ IDE

2.2 Configuring the Application

1. Open ***sapis/examples/wlan/smtp_client/rsi_smtp_client_app.c*** file and update/modify following macros.

SSID refers to the name of the Access point.

```
#define SSID                    "<ap name>"
```

CHANNEL_NO refers to the channel in which device should scan. If it is 0, device will scan all channels.

```
#define CHANNEL_NO          0
```

SECURITY_TYPE refers to the type of security. In this application STA supports Open, WPA-PSK, WPA2-PSK securities.

Valid configuration is:

RSI_OPEN - For OPEN security mode

RSI_WPA - For WPA security mode

RSI_WPA2 - For WPA2 security mode

```
#define SECURITY_TYPE           RSI_OPEN
```

PSK refers to the secret key if the Access point configured in WPA-PSK/WPA2-PSK security modes.

```
#define PSK                     "<psk>"
```

To configure SMTP client Parameters

To select IPv6, FLAGS should be set to 1, by default it supports IPv4

```
#define FLAGS           0
```

To configure the SMTP server port, default port is 25. If server can listen on other non standard ports, change the SMTP_PORT

```
#define SMTP_PORT       25
```

To configure the username for authentication

```
#define USERNAME        "username"
```

To configure the password for authentication

```
#define PASSWORD        "password"
```

To configure the sender's address

```
#define FROM_ADDRESS      "abc@mail.local"
```

IP address of the SMTP server should be in long format and in little endian byte order.

Example: To configure "192.168.0.2" as IP address, update the macro **SERVER_IP** as **0x010AA8C0**.

```
#define SERVER_IP          0x0200A8C0
```

To configure the authentication type

Supported types are,

`RSI_SMTP_CLIENT_AUTH_LOGIN (1)` – for login type

`RSI_SMTP_CLIENT_AUTH_PLAIN (3)` – For plain authentication

```
#define AUTH_TYPE         RSI_SMTP_CLIENT_AUTH_LOGIN
```

To configure the priority

Supported priorities are,

`RSI_SMTP_MAIL_PRIORITY_LOW (1)` - For Low priority

`RSI_SMTP_MAIL_PRIORITY_NORMAL (2)` - For Normal Priority

`RSI_SMTP_MAIL_PRIORITY_HIGH (4)` - For High Priority

```
#define PRIORITY          RSI_SMTP_MAIL_PRIORITY_NORMAL
```

To configure domain name of the client.

```
#define CLIENT_DOMAIN      "mymail.mail.com"
```

To configure mail recipient address.

```
#define MAIL_RECIPIENT_ADDRESS    "test@mail.local"
```

To configure subject of the mail. This should be a string

```
#define MAIL_SUBJECT      "TEST"
```

To configure mail body.

```
#define MAIL_BODY          "TEST BODY"
```

Note:

Mail body to the mail send API (rsi_smtp_client_mail_send_async) need not be a string. A buffer with definite length can be pointed to the API.

To configure IP address

DHCP_MODE refers whether IP address configured through DHCP or STATIC

```
#define DHCP_MODE    1
```

Note: If user wants to configure STA IP address through DHCP then set **DHCP_MODE** to 1 and skip configuring the following **DEVICE_IP, GATEWAY** and **NETMASK** macros.
(Or)
If user wants to configure STA IP address through STATIC then set **DHCP_MODE** macro to "0" and configure following **DEVICE_IP**, **GATEWAY** and **NETMASK** macros.

IP address to be configured to the device in STA mode should be in long format and in little endian byte order.

Example: To configure "192.168.10.10" as IP address, update the macro **DEVICE_IP** as **0x0A0AA8C0**.

```
#define  DEVICE_IP          0X0A0AA8C0
```

IP address of the gateway should also be in long format and in little endian byte order

Example: To configure "192.168.10.1" as Gateway, update the macro GATEWAY as **0x010AA8C0**

```
#define  GATEWAY            0x010AA8C0
```

IP address of the network mask should also be in long format and in little endian byte order

Example: To configure "255.255.255.0" as network mask, update the macro **NETMASK** as **0x00FFFFFF**

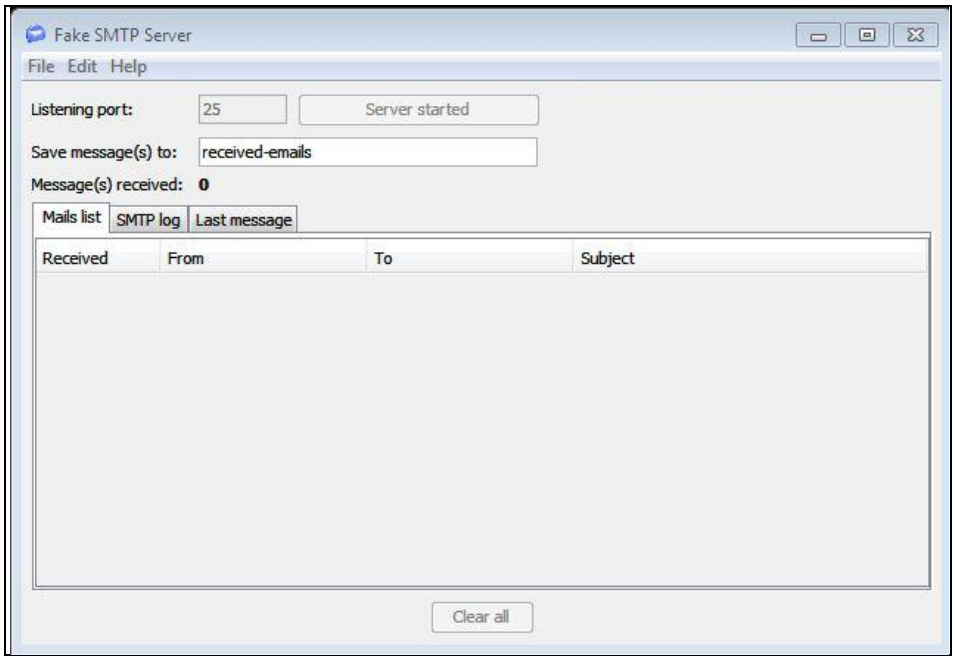
```
#define   NETMASK           0x00FFFFFF
```

2. Open ***sapis/include/rsi_wlan_config.h*** file and update/modify following macros,

```
#define  CONCURRENT_MODE             RSI_DISABLE
#define  RSI_FEATURE_BIT_MAP         FEAT_SECURITY_OPEN
#define  RSI_TCP_IP_BYPASS           RSI_DISABLE
#define RSI_TCP_IP_FEATURE_BIT_MAP  (TCP_IP_FEAT_DHCPV4_CLIENT
                                   | TCP_IP_FEAT_SMTP_CLIENT)
#define  RSI_CUSTOM_FEATURE_BIT_MAP  0
#define  RSI_BAND                    RSI_BAND_2P4GHZ
```

2.3 Executing the Application

1. Configure the Access point in OPEN/WPA-PSK/WPA2-PSK mode to connect WiSeConnect device in STA mode.
2. Download SMTP server and run SMTP server in Windows PC2.



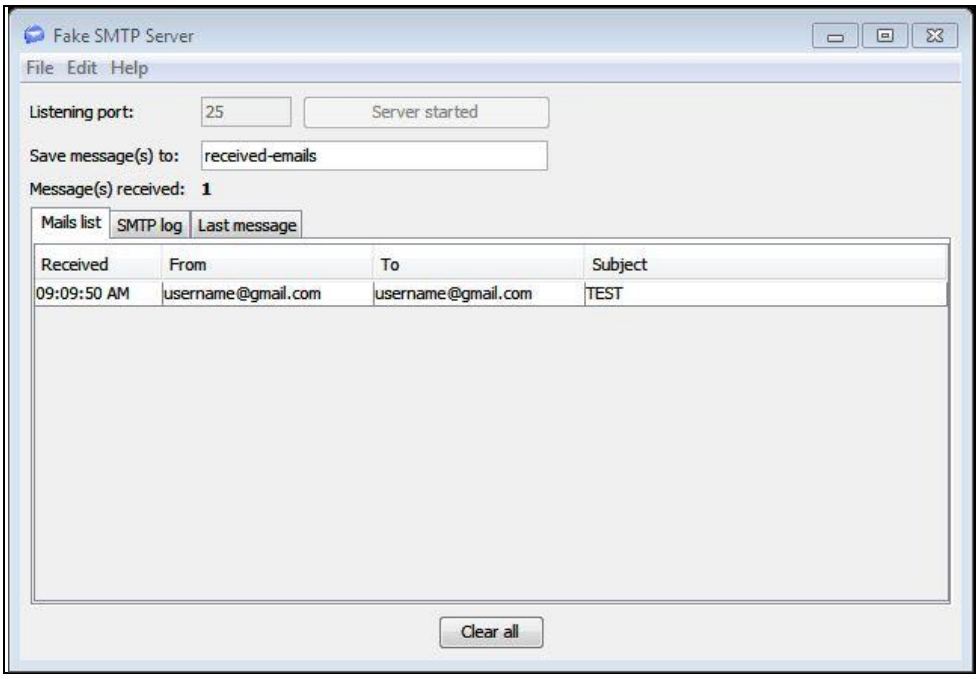
3. **SPI Interface**

   If User using SPI interface, Please refer the document ***sapis/platforms/spansion_MB9BF568NBGL/RS9113-WiSeConnect_SAPIS_Spansion_Project_User_guide.pdf*** for executing the ***smtp_client*** example in CooCox IDE.

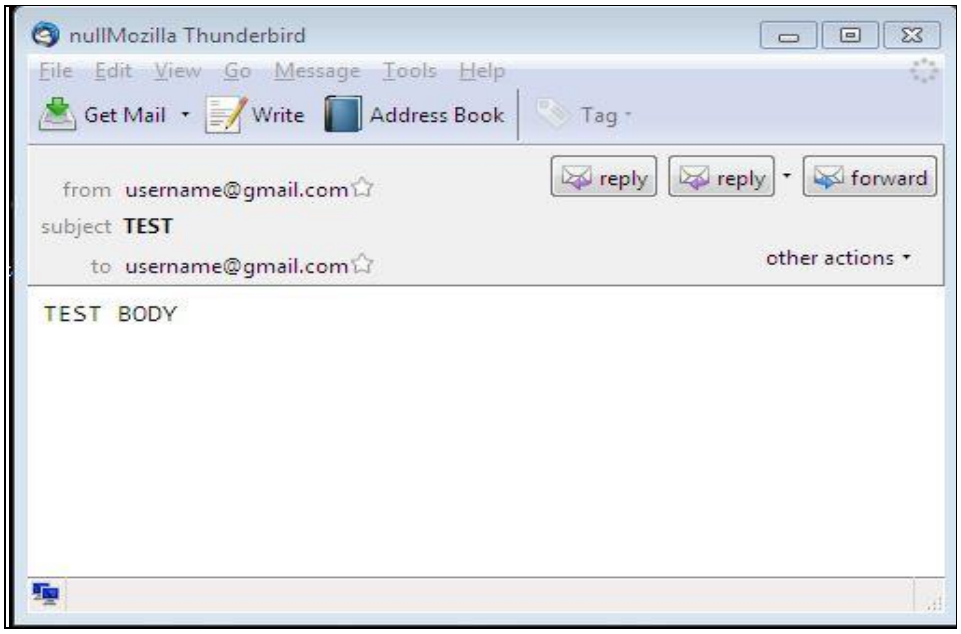
4. **UART/USB-CDC Interface**

   If User using UART interface, Please refer the document ***sapis/platforms/windows_uart/RS9113-WiSeConnect_SAPIS_Windows_Project_UserGuide.pdf*** for executing the ***smtp_client*** example in Dev-C++ IDE

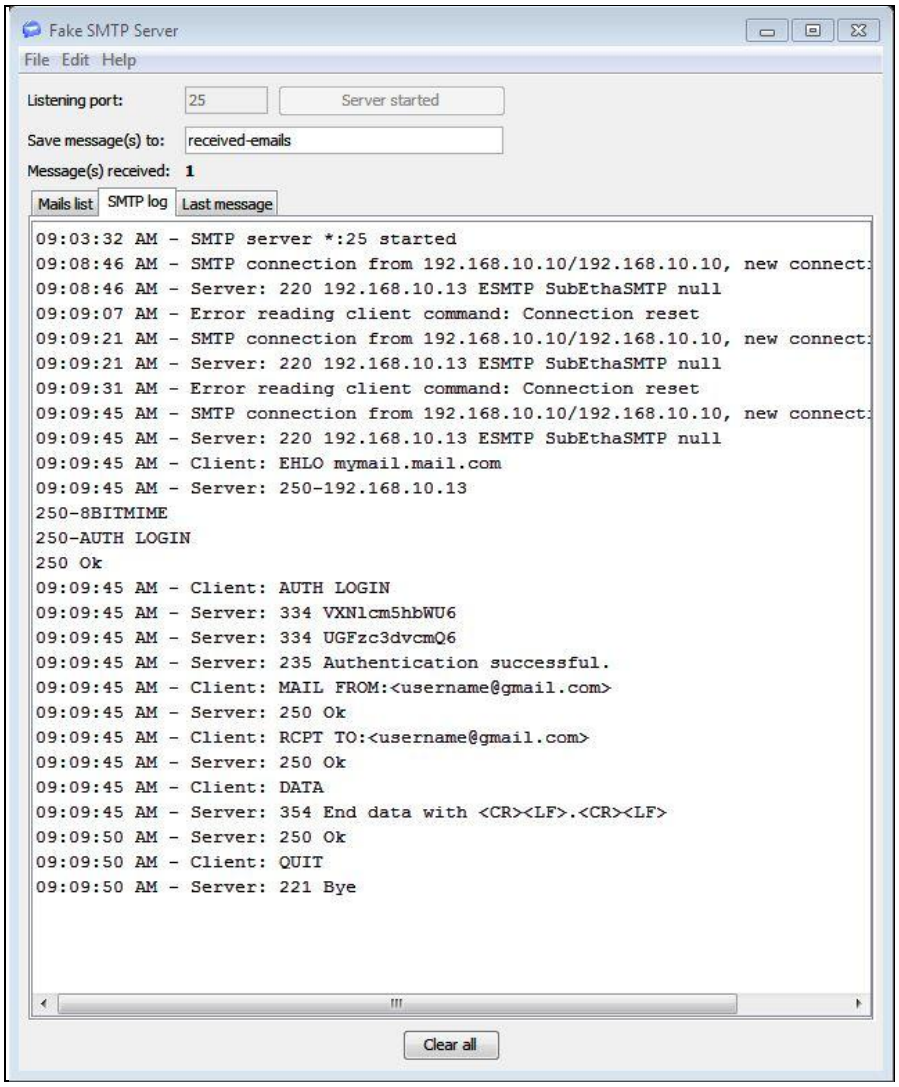
5. After the program gets executed, WiSeConnect Device would be connected to Access point and get IP.
6. After successful connection with Access Point, Device starts authenticating with the SMTP server running on Windows PC2.
7. After successful authentication, it sends a mail to the server.
8. Please refer the given below images for SMTP server receives mail sent by WiSeConnect device.



9. Double click the mail and check mail subject and mail body sent by WiSeConnect Device.



10. Check the log messages at SMTP server


Fake SMTP Server
File Edit Help
Listening port: 25 Server started
Save message(s) to: received-emails
Message(s) received: 1
Mails list SMTP log Last message
09:03:32 AM - SMTP server *:25 started
09:08:46 AM - SMTP connection from 192.168.10.10/192.168.10.10, new connect
09:08:46 AM - Server: 220 192.168.10.13 ESMTP SubEthaSMTP null
09:09:07 AM - Error reading client command: Connection reset
09:09:21 AM - SMTP connection from 192.168.10.10/192.168.10.10, new connect
09:09:21 AM - Server: 220 192.168.10.13 ESMTP SubEthaSMTP null
09:09:31 AM - Error reading client command: Connection reset
09:09:45 AM - SMTP connection from 192.168.10.10/192.168.10.10, new connect
09:09:45 AM - Server: 220 192.168.10.13 ESMTP SubEthaSMTP null
09:09:45 AM - Client: EHLO mymail.mail.com
09:09:45 AM - Server: 250-192.168.10.13
250-8BITMIME
250-AUTH LOGIN
250 Ok
09:09:45 AM - Client: AUTH LOGIN
09:09:45 AM - Server: 334 VXNlcm5hbWU6
09:09:45 AM - Server: 334 UGFzc3dvcmQ6
09:09:45 AM - Server: 235 Authentication successful.
09:09:45 AM - Client: MAIL FROM:<username@gmail.com>
09:09:45 AM - Server: 250 Ok
09:09:45 AM - Client: RCPT TO:<username@gmail.com>
09:09:45 AM - Server: 250 Ok
09:09:45 AM - Client: DATA
09:09:45 AM - Server: 354 End data with <CR><LF>.<CR><LF>
09:09:50 AM - Server: 250 Ok
09:09:50 AM - Client: QUIT
09:09:50 AM - Server: 221 Bye
Clear all