

CC31xx NWP Filter Application

Overview

This is a sample application demonstrating how NWP filters can be added to CC3100 for managing the Rx filtering process.

The Rx-Filters feature enables the user to simply define, and manage the Rx-filtering process. It reduces the amount of traffic transferred to the host, and achieves an efficient power management.

The host-driver supplies a set of APIs to enable filter creation\editing. The APIs include the following operations:

- Add nodes: The user may request to add new node to the database, the node is created and a node unique ID is returned back to the user. The user should disable all filters before adding a node.
- Remove nodes: The user may request to remove a node from the filters' database, the node is removed. The user should disable all filters before removing a node.
- Enable nodes: The user can request to enable one or more filters, filters which are enabled, take part in the matching process.
- Disable nodes: The user can request to disable one or more filters, filters which are disabled, don't take part in the matching process.
- Get filter node
- Update filter node

The application is responsible for adding filter nodes and defining the filters' tree hierarchy. If the user is interested in storing a filter node in the flash memory, he should set the persistence flag of the node. A filter can be defined as persistent only if its parent is persistent. Persistent filter nodes will be loaded automatically upon system startup.

The application developer can choose one or more filters to disable\enable, filters which are disabled don't take part in the matching process.

More information in **Rx-Filters** section of [Advanced_Networking_Document](#)

Note: This wiki page is only applicable for **CC3100-SDK v1.0.0** and upward releases. For documentation on older SDKs' examples, refer corresponding file in `<cc3100-sdk-installation-location>\cc3100-sdk\docs\examples\`

Application details

- The application creates and enables two filters to filter packets according to:
 1. Remote MAC address
 2. Remote IP address
- It then connects to an AP and creates a TCP server waiting for data on PORT_NUM. The connection attempts from the remote devices whose IP/MAC addresses matches the filtered addresses are rejected

[Return to CC31xx & CC32xx Home Page](#)[Return to CC31xx Sample Applications](#)

Usage

- Connect the board to a Windows-PC and configure the terminal-program for seeing the logs - [CC31xx_&_CC32xx_Terminal_Setting_Wiki^[1]] has detailed instructions for configuring the terminal-program
- Edit **sl_common.h** and update the following parameters for the device to establish connection w/ the access-point

```
#define SSID_NAME      "<ap_name>"
#define SEC_TYPE       <security-type>
#define PASSKEY        " "
```

- Edit **main.c** and update the following parameters for the device to be filtered

```
UINT8 g_MacAddress[SL_MAC_ADDR_LEN] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
UINT8 g_IpAddress[IP_ADDR_LENGTH] = {0x00, 0x00, 0x00, 0x00};
```

- Build and launch the application

```
Both the filters describe above will be created and enabled
The device will connect to an AP, create a socket-server (TCP) and wait for connections on PORT_NUM
```

- Open an 'Iperf' client on Windows-PC and connect on 'PORT_NUM'

```
Iperf -c <DEST_IP_ADDR> -p <PORT_NUM> -i 1
```

- If the IP/MAC address of the Windows-PC (on which IPerf is running) matches the filtered addresses, the connection requests will not be accepted by CC3100.
- See the self explanatory logs on the terminal-program's console.

Limitations/Known Issues

- Payload rule is currently not supported
- Filter's action of turning on/off GPIO upon a match is currently not supported
- Filter's action of sending EVENT to the host upon a match is currently not supported
- BSSID can't be filtered while STA is connected to an AP, Adding BSSID filter while STA is connected will cause disconnection.

References

[1] http://processors.wiki.ti.com/index.php/CC31xx_&_CC32xx_Terminal_Setting

Article Sources and Contributors

CC31xx NWP Filter Application *Source:* <http://processors.wiki.ti.com/index.php?oldid=189802> *Contributors:* A0131814, A0132173, A0221015, Codycooke, Malokyle

Image Sources, Licenses and Contributors

File:Cc31xx cc32xx return home.png *Source:* http://processors.wiki.ti.com/index.php?title=File:Cc31xx_cc32xx_return_home.png *License:* unknown *Contributors:* A0221015

File:Cc31xx return sample apps.png *Source:* http://processors.wiki.ti.com/index.php?title=File:Cc31xx_return_sample_apps.png *License:* unknown *Contributors:* A0221015