FIPS MODE SUPPORT

Application Note *v1.4*

INTRODUCTION

The goal of this document includes the following:

- Explain how to enable FIPS Mode on the Laird WB45NBT.
- Describe the build process for including FIPS support.
- Offer practical notes regarding FIPS mode support.

OVERVIEW

FIPS (Federal Information Processing Standard) is a National Institute of Standards and Technology (NIST) standard for data used in computer systems. Of the most interest to Wi-Fi systems is FIPS 140-2, which defines security requirements for cryptographic modules, or modules that provide security services such as encryption to provide data.

FIPS support is provided by **openssl_fips-2.0** with the SDC supplicant and additional kernel and user space components.

FIPS is enabled with an option (**-F**) to the supplicant and in the driver by setting the fips_mode=y option and loading additional kernel/user space components. The wireless init script (/etc/network/wireless.sh) starts the supplicant, driver, and additional kernel/user space components as required for FIPS and non-FIPS modes of operation.

When running in FIPS mode, the WB45NBT radio will only use AES-CCMP-128 encryption. In FIPS mode, the supplicant will restrict operation to WPA2-AES with EAP-TLS.

If no encryption key has been established, i.e. during authentication, 802.1x/eapol authentication packets are transmitted and received unencrypted, and all other transmit and receive packets are discarded.

METHOD OF ENABLING FIPS MODE

Note: The wireless init-script (/etc/network/wireless.sh) handles stop, start, and restart of Wi-Fi, by loading the driver along with additional kernel/user space components and then starting the supplicant. This should be executable as a system command.

The wireless interface must be restarted for FIPS mode to take effect.

The following are methods for enabling FIPS-mode.

- By setting the mode via SDK:
  To programmatically control the FIPS mode from a host, use the SDK functions GetGLobalSettings, change the value, and then SetGlobalSettings.

  Code snippet:

```
SDCGlobalConfig global;
GetGlobalSettings(&global);
global.suppInfo |= 1;
SetGlobalSettings(&global);
```

  From the Linux SDK header file, the bits are defined as:

```
unsigned long suppInfo;   //bit 0 is Summit FIPS on/off, bit 1 is WAPI on/off
```

- By setting the mode via **sdc_cli**.
  The wireless init-script first checks the global profile setting.

```
# sdc_cli global set fips enable
```

- Directly via command-line (not-persistent across reboots):

```
# wireless fips start
```

- By permanently enabling FIPS-mode in the init script (/etc/network/wireless.sh). To enable, uncomment the following line:

```
#WIFI_FIPS=-F
```

Note: The wireless can be restarted into or out of fips-mode depending on either the set configuration or the command-line. If the FIPS support modules and user application is seen listed, then FIPS mode is active. Use the following command to view the current status:

```
# wireless
```

BUILDING AND INCLUDING FIPS SUPPORT

Check the **buildroot** configuration options to enable building and optionally including FIPS. The default is to normally build with fips-mode support, but not necessarily include all components on the target.

- The **openssl** package must be built using **openssl_fips** (which gets built first).
- The **sdcsupp** package is built using **openssl with fips support**.
- The **openssl_fips-2.0** package is provided as a certified copy of *OpenSSL FIPS 2.0* from OpenSSL (http://www.openssl.org). A trusted path copy of OpenSSL FIPS 2.0 is required to be compliant. Refer to OpenSSL FIPS documentation for additional information.
- Additionally, driver FIPS support is provided by the following at run time:
  - ath6kl_laird.ko
  - sdc2u.ko
  - sdcu (user space application)

Notes Regarding OpenSSL:

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)

This product includes cryptographic software written by Eric Young. (eay@cryptsoft.com)

This product includes software written by Tim Hudson. (tjh@cryptsoft.com)

SOME PRACTICAL NOTES

The following are some practical notes regarding FIPS mode support:

- FIPS mode only works with WPA2 AES EAP-TLS.
- PKCS#12 files don't work in FIPS mode.
- You must use PEM files for cert/keys.
- If you attempt to use a configuration other than WPA2-AES/EAP-TLS and debug output is enabled, the following error messages are output:
  - "CFG: Disabled. Invalid WPA type for FIPS."
  - "CFG: Disabled. Invalid EAP type for FIPS."
- The private key can be encrypted or unencrypted. However, if the private key is encrypted, it must be of PKCS#8 format. If the private key starts with the following, it is the old format (which uses MDS) and is not FIPS compliant You may use OpenSSL to fix the certificate in these cases by converting it to PKCS12 and then back to PEM as follows:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,2FA91DAE900C159E
```

- OpenSSL can be used to fix the certificate by converting to PKCS12 and then back to PEM using a script as follows:

```
openssl pkcs12 -export -out $1.p12 -inkey $1 -in $1 -passin pass:$2 -passout pass:$2
openssl pkcs12 -in $1.p12 -out $1.new -passin pass:$2 -passout pass:$2
```

- Use the following to see if the drivers are in FIPS mode:

```
cat /sys/module/ath6kl_core/parameters/fips_mode
```

- The supplicant is in FIPS mode if it has been executed with -F.
  To look for "-F" in the supplicant startup, run the following command:

```
ps | egrep sdcsupp
```

REVISION HISTORY

Revision	Date	Description	Approved By
1.0	22 July 2013	Initial Release	Steve DeRosier
1.1	28 July 2013	Further Edits	Steve DeRosier
1.2		Not Published	
1.3		Not Published	
1.4	24 Oct 2013	Further Edits	John Imboden