



ADEPT™-WFA Compliance Tester

**Azimuth Systems, Inc.
31 Nagog Park
Acton, MA 01720
Tel. 978.263.6610
Fax 978.263.5352
www.azimuthsystems.com**

Copyright © 2006 Azimuth Systems, Inc. All rights reserved. Printed in United States of America.

Azimuth, Azimuth DIRECTOR, SpyNIC, testMAC, ADEPT and ACE are trademarks of Azimuth Systems, Inc. Microsoft and Windows are trademarks of Microsoft Corporation. Adobe, Acrobat, and Acrobat Reader are trademarks of Adobe Systems Incorporated. All other third-party trademarks and service marks referred to in these materials are the property of their owners. No part of this documentation 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 Azimuth Systems, Inc. Azimuth Systems, Inc. provides this documentation "AS IS," without warranty, term, or condition of any kind, either implied or expressed, including, but not limited to, the implied warranties, terms, or conditions of merchantability, satisfactory quality, non-infringement and fitness for a particular purpose. Azimuth Systems, Inc. reserves the right to make changes to equipment design or program components described in this documentation, as progress in engineering, manufacturing methods, or other circumstances may warrant. No responsibility is assumed for the use of Azimuth Systems, Inc. software or hardware, all rights, obligations and remedies related to which are as set forth in the applicable sales and license agreements.

Azimuth Systems, Inc.
31 Nagog Park
Acton, MA 01720
Tel. 978.263.6610
Fax 978.263.5352
www.azimuthsystems.com
Technical Publications Doc. No. 101006, Rev. v2.1
Published 10/10/06

Electromagnetic Emissions Statements for Azimuth ADEPT-WFA Instruction/Installation Manuals:

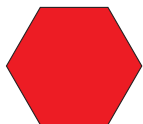
USA Requirements: Federal Communications Commission (FCC) Compliance Notice: Radio Frequency Notice

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy. If it is not installed and used in accordance with the instruction manual, it may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to take whatever measures may be necessary to correct the interference at their own expense.

Canada Requirements: Canadian Department of Communications Radio Interference Regulations

This digital apparatus does not exceed the Class A limits for radio-noise emissions from digital apparatus as set out in the Radio Interference Regulations of the Canadian Department of Communications.

Règlement sur le brouillage radioélectrique du ministère des Communications Le present appareil numerique n'emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de la class A prescrites dans le Reglement sur le brouillage radioelectrique edicte par le ministere des Communications du Canada.



Caution: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Table of Contents

The ADEPT-WFA Compliance Tester

- Device Management 1-1
 - Assigning a Management IP Address for the Control PC 1-1
 - Adding/Managing the ADEPT-WFA's Test Engines 1-2
 - Modifying DHCP Settings 1-4
 - Upgrading the ADEPT-WFA Firmware 1-5
 - Updating the ADEPT-WFA License Key 1-6
 - Restarting the ADEPT-WFA Test Engine 1-7
 - Modifying Test Engine Parameters 1-8
 - Obtaining Properties of the ADEPT-WFA Test Engine 1-9
 - Viewing and Configuring Logs 1-10
- WiFi Compliance Tests 1-11
 - Configuring and Running a NAV Test 1-11
 - Configuring and Running a PLCP Test 1-14
 - Configuring and Running a MIC Test 1-17
 - Configuring/Running the STAUT MIC Test 1-18
 - Configuring/Running the APUT MIC Test 1-23
 - 1-28

Appendix A: ADEPT-WFA Compliance Tester Tcl Commands

- Initiating a Tcl Session A-2
- Running the NAV Test A-3
- Running the PLCP Test A-3
- Running the STAUT MIC Countermeasure Test A-4
- Running the APUT MIC Countermeasure Test A-7

Appendix B: ADEPT-WFA Tcl Commands

Index

The ADEPT-WFA Compliance Tester

Launch the ADEPT-WFA Compliance Tester by double-clicking the associated desktop icon on the control PC.

Device Management

This section covers Compliance Tester requirements and options that need to be set up previous to running compliance tests.

Assigning a Management IP Address for the Control PC

The ADEPT-WFA Compliance Tester allows you to manage your ADEPT-WFA from any network interface on the control PC. The Select Connection Interface tab allows you to assign the IP address from the control PC to the Adept-WFA. (Figure 1-1).

Note: The Bus-Net IP address must be on a different subnet than any IP address of the test bed equipment



Figure 1-1. Select Connection Interface Tab

To Assign a Management IP Address for the Control PC:

1. Select from among the list of IP addresses that are in the **IP Address** drop-down menu. The IP addresses available in the drop-down menu are addresses associated with NICs on the PC that the Compliance Tester detects.
2. Click [Next>>]
The ADEPT-WFA Explorer tab displays.

Adding/Managing the ADEPT-WFA's Test Engines

Add an ADEPT-WFA's Test Engines to the ADEPT-WFA Compliance Tester at the ADEPT WFA Explorer tab (Figure 1-2).

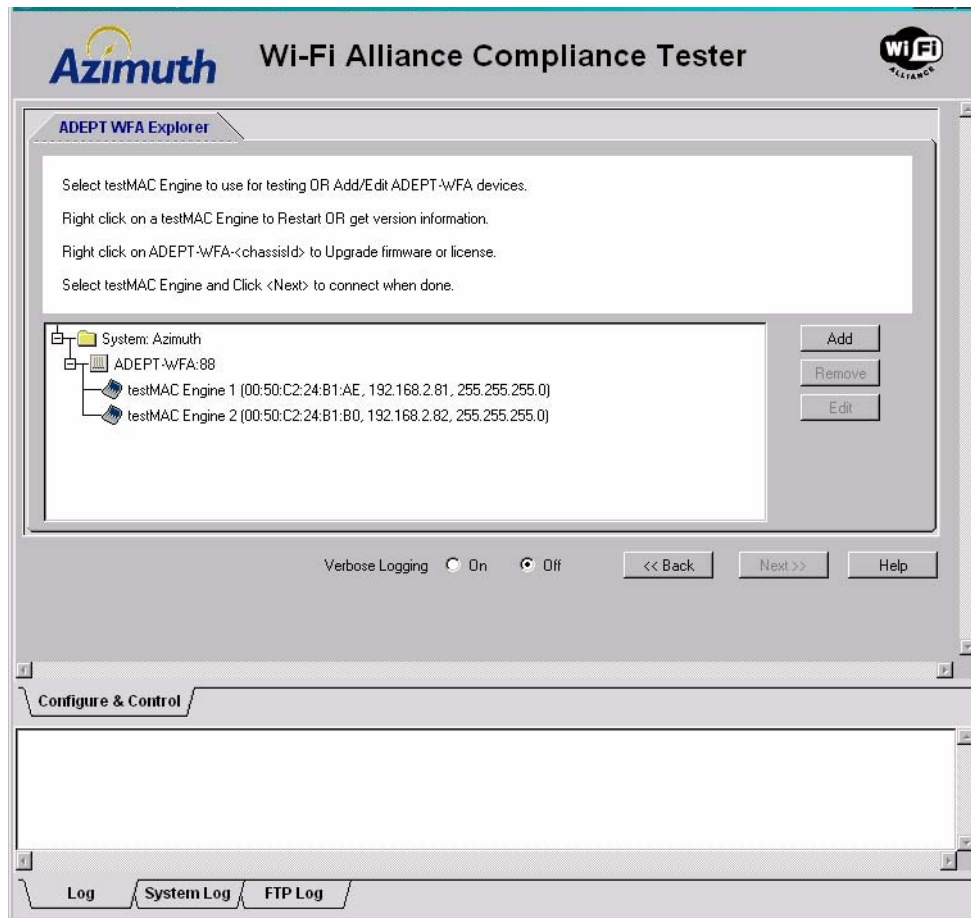


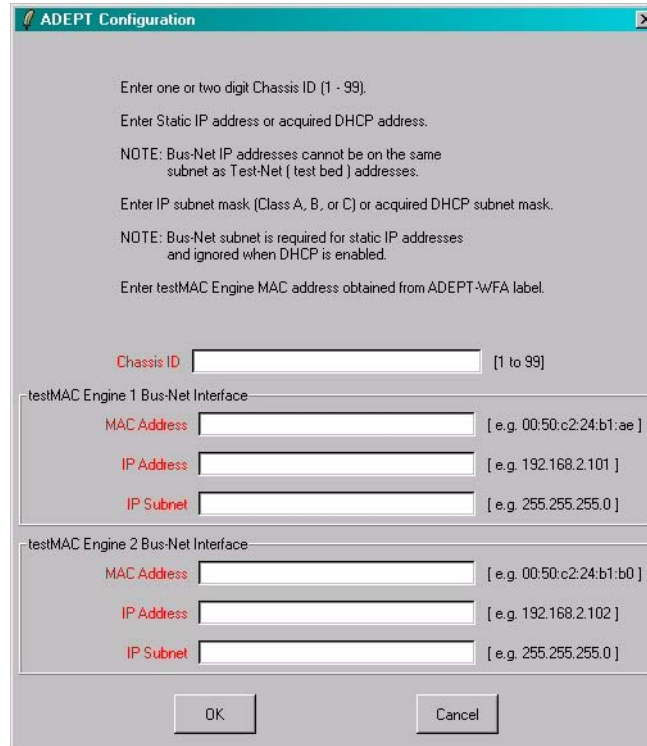
Figure 1-2. ADEPT WFA Explorer Tab

The ADEPT WFA Explorer tab features an explorer tree that allows you to expand and collapse branches to view/access the Test Engine(s) once you add the ADEPT-WFA.

Note: The Bus-Net IP address must be on a different subnet than any IP address of the test bed equipment.

To add the ADEPT-WFAs' Test Engines:

1. Click [Add] on the ADEPT WFA Explorer tab. The ADEPT Configuration screen displays (Figure 1-3).



The image shows a dialog box titled "ADEPT Configuration". It contains several text input fields and instructions. At the top, it says "Enter one or two digit Chassis ID (1 - 99)". Below that is a field for "Chassis ID" with a range "[1 to 99]". Next is "Enter Static IP address or acquired DHCP address." followed by a "NOTE: Bus-Net IP addresses cannot be on the same subnet as Test-Net (test bed) addresses." Then "Enter IP subnet mask (Class A, B, or C) or acquired DHCP subnet mask." followed by another "NOTE: Bus-Net subnet is required for static IP addresses and ignored when DHCP is enabled." Finally, "Enter testMAC Engine MAC address obtained from ADEPT-WFA label." Below these are two sections for test engines. The first is "testMAC Engine 1 Bus-Net Interface" with fields for "MAC Address" (example: 00:50:c2:24:b1:ae), "IP Address" (example: 192.168.2.101), and "IP Subnet" (example: 255.255.255.0). The second is "testMAC Engine 2 Bus-Net Interface" with fields for "MAC Address" (example: 00:50:c2:24:b1:b0), "IP Address" (example: 192.168.2.102), and "IP Subnet" (example: 255.255.255.0). At the bottom are "OK" and "Cancel" buttons.

Figure 1-3. ADEPT Configuration Dialog Box

2. Assign a 1-2 digit integer in the **Chassis ID** field. This is an identification number for the ADEPT-WFA that appears in the chassis ID display on the front panel of the ADEPT-WFA after communications are established.
3. Enter the MAC address of Test Engine 1 and Test Engine 2 in their respective fields in the format `xx:xx:xx:xx:xx:xx`. The MAC addresses for the network interface associated with Test Engine 1 and Test Engine 2 are printed on a label affixed adjacent to the power switch on the back of the unit as illustrated in Figure 1-3.
4. Enter the IP address and IP subnet of Test Engine 1 and Test Engine 2 in their respective fields in the format `nnn.nnn.nnn.nnn`, replacing the *n*'s with the numbers in the address. If you are running a DHCP server, obtain the Test Engine IP addresses from the DHCP server. If you are not running a DHCP server, you can assign a static IP address in this field.
Note: A value is required in the IP subnet field of the ADEPT configuration dialog box even if you are using a DHCP server to obtain IP addresses. When DHCP is enabled, the IP subnet you enter will be ignored.
5. Click [OK]. The ADEPT-WFA with test engines is added to the explorer tree of the ADEPT WFA Explorer tab (Figure 1-2).

To manage the ADEPT-WFA's test engines:

1. To edit the configuration of an ADEPT-WFA's Test Engine(s), select the ADEPT-WFA from the tree on the ADEPT WFA Explorer tab and click [Edit]. The ADEPT Configuration screen displays (Figure 1-3). Make the necessary edits to the configuration of each test engine and click [OK] to accept the changes and close the dialog box.
2. Select a test engine (i.e., Test Engine 1 or Test Engine 2).
3. Click [Next>>] to connect to the ADEPT-WFA and display the Tests configuration tab. If you entered a new static IP address or edited an existing one, you are prompted to verify setting the Test Engine IP addresses to the values in the selected configuration. If so, click [OK] to accept the settings.
4. If you need to modify the Test Engine's operating band, channel, SSID, or PSK for any of the NAV/PLCP/MIC tests, please see "[MIC APUT Failed: No ping response received. After first corrupt data frame and 21 seconds elapsed.](#)" (page 1-28).

You are now ready to begin configuring and running the NAV/PLCP/MIC tests.

To remove an ADEPT WFAs' test engines, select the engine(s) on the ADEPT WFA Explorer tab and click [Remove]. The selected device disappears from the explorer tree.

Modifying DHCP Settings

The following procedure describes how to change DHCP settings, used to specify whether the Test Engine IP address is static or assigned by a DHCP server. Please note the following:

- DHCP is shipped disabled.
- If you want to modify this setting, you can configure the DHCP setting for each Test Engine after you initially configure the Test Engine IP addresses as described in "[Adding/Managing the ADEPT-WFA's Test Engines](#)" (page 1-2).
- The DHCP settings for each Test Engine must be managed separately.

Note: The Bus-Net IP address must be on a different subnet than any IP address of the test bed equipment.

To Modify DHCP Settings:

1. Select the Advanced tab (Figure 1-4) to change DHCP settings.

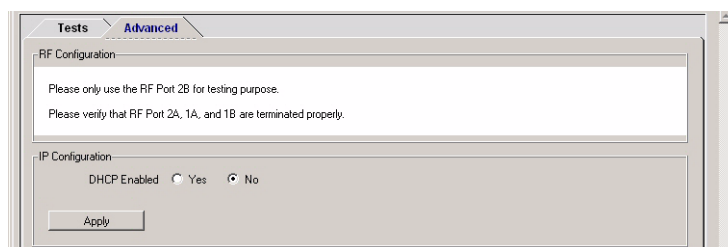


Figure 1-4. Advanced Tab

2. If you are running a DHCP server, select the **Yes** radio button. If you are not running a DHCP server, select the **No** radio button.
3. Click [Apply]. The ADEPT-WFA reboots.

If you disabled DHCP, a prompt appears with wording similar to the following: *Please find the IP address assigned by the DHCP server to the test engine. Then edit the properties in the next screen to enter the IP address*

If you enabled DHCP, a prompt appears with wording similar to the following: *Please find the static IP address of the test engine. Then edit the properties in the next screen to enter the IP address.*

4. Click [OK] to close the prompt dialog box.
Note: If you are setting or changing a static IP address, a screen appears when connecting for the first time with wording similar to the following: *Do you want to set Test Engine (MAC address) to IP address (nnn.nnn.nnn.nnn)?* Click [OK] to accept the change.
5. The ADEPT WFA Explorer tab displays (see [Figure 1-2](#)).
6. Select the ADEPT-WFA in the Explorer tab and click [Edit]. The ADEPT Configuration screen displays (see [Figure 1-3](#)).
7. Enter the appropriate IP address and IP subnet (static or assigned by the DHCP server) for each Test Engine.
8. Click [OK].
9. Click [Next>>] to advance to the Tests tab.
10. Repeat this procedure for the other Test Engine.

Upgrading the ADEPT-WFA Firmware

The ADEPT-WFA firmware should be upgraded every time the ADEPT-WFA Compliance Tester software is upgraded; software must be upgraded before firmware. For software upgrade information, reference the ADEPT-WFA Compliance Tester Installation and Upgrade guide.

To Upgrade the ADEPT-WFA Firmware:

1. Start the ADEPT-WFA Compliance Tester and enter the appropriate management IP address for the control PC.
2. Click [Next>>]. The ADEPT WFA Explorer tab appears.
3. Right-click the ADEPT-WFA device in the explorer tree and select **Upgrade** from the menu that displays ([Figure 1-5](#)). The Input Upgrade Image screen displays ([Figure 1-6](#)).

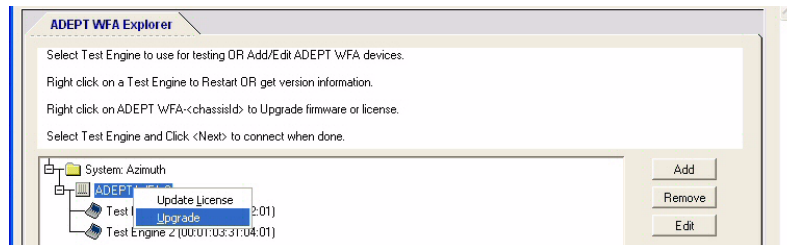


Figure 1-5. Upgrade Option

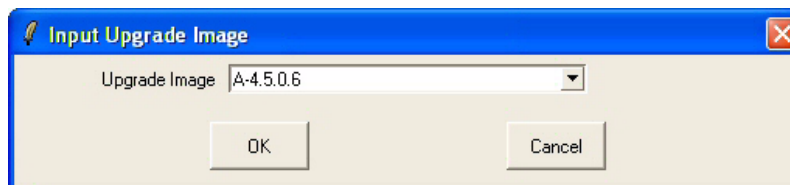


Figure 1-6. Input Upgrade Image Dialog Box

4. Select the appropriate version number from the Upgrade Image drop-down menu.
5. Click [OK]. The selected upgrade image is downloaded to the ADEPT-WFA firmware of Test Engine 1, and then that station is rebooted. Then the image is downloaded to Test Engine 2, which is rebooted when the download is complete. Messages on the success of this operation are available on the Log tab.

Note: The upgrade procedure can take several minutes. It is important not to interrupt the upgrade process once it has started.

Updating the ADEPT-WFA License Key

The ADEPT-WFA offers many optionally licensed features for use in Wi-Fi testing and analysis, including emulating stations (called *softClients*) and access points (called *softAPs*) and capturing packets for use in traffic analysis (called *Packet Capture*). You can also expand and control the functionality of the ADEPT-WFA through the use of its Tcl-based Programmable Extension (called *PE*). Each of the optional ADEPT-WFA capabilities can be unlocked through the purchase of an additional license. Separate licenses for using the *softClient*, *softAP*, *Packet Capture* and *PE* capabilities can be obtained through your Azimuth Sales representative. The following procedure describes how to add a license to the ADEPT-WFA. For more information about each of the optional licensed features, please see [“Optional ADEPT-WFA Wi-Fi Capabilities”](#) (page 1-3).

To Add a License to the ADEPT-WFA:

1. Start the ADEPT-WFA Compliance Tester and enter the appropriate management IP address for the control PC.
2. Click [Next>>]. The ADEPT WFA Explorer tab appears.

3. Right-click the ADEPT WFA device in the explorer tree of the ADEPT WFA Explorer tab and select **Update License** from the menu that displays (Figure 1-7). The Input License Key screen displays (Figure 1-8).

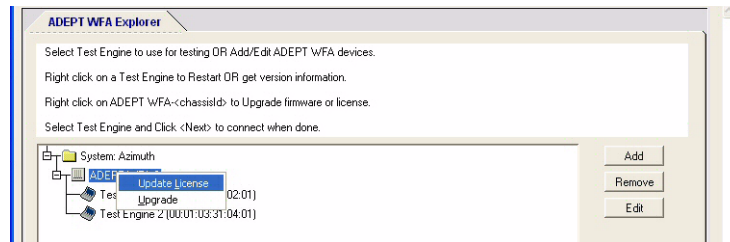


Figure 1-7. Update License Option

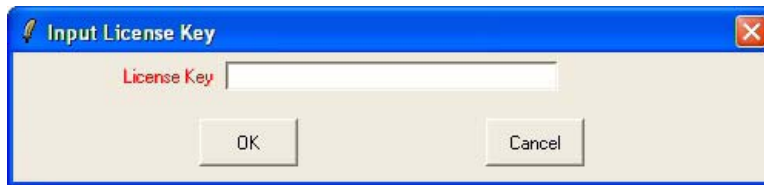


Figure 1-8. Input License Key Dialog Box

4. Enter the license key for the additional functionality in the License Key field.
5. Click [OK]. You can now use the additional functionality associated with the license key that you entered.

Restarting the ADEPT-WFA Test Engine

You can restart Test Engine 1 or Test Engine 2 separately, without having to power cycle the ADEPT-WFA.

To restart an ADEPT-WFA test engine:

1. Start the ADEPT-WFA Compliance Tester and enter the appropriate management IP address for the control PC.
2. Click [Next>>]. The ADEPT WFA Explorer tab displays.
3. In the explorer tree of the ADEPT WFA Explorer tab, right-click the Test Engine that you want to restart and select **Restart** from the menu that displays (Figure 1-9). Messages are available on the Log tab that indicate a test engine reboot is taking place and when the test engine is available again.

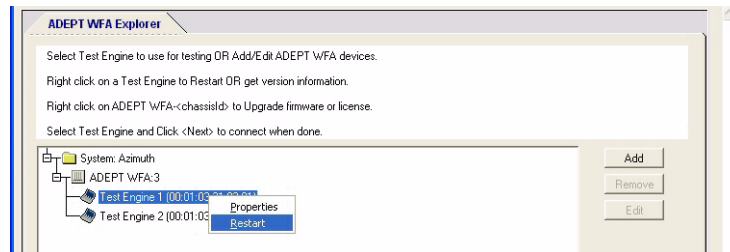


Figure 1-9. Restart Option

Modifying Test Engine Parameters

A *wifi_config_te<x>.tcl* (where $x = 1$ or 2) configuration file on the control PC is used to define several of the Wi-Fi parameters for the Test Engine. This file is automatically generated the first time you select a test engine on the ADEPT-WFA Explorer tab and click [Next>>]; the Tests tab displays and a configuration file is written to the following path:

`c:\Program Files\Azimuth\ADEPT-WFA\data\config\wifi_config_te1.tcl`

The values written to the file are the test parameters as defined in the WFA test specifications and are used as the default values for the test engine.

If necessary, you can edit any of the following parameters in the configuration file:

- Wi-Fi channel number at each operating band (a, b, and g) for each test (NAV, PLCP and MIC Countermeasures).
- SSID and PSK values for WPA or WPA2 MIC Countermeasure tests.

To restore the default *wifi_config_te<x>.tcl* (where $x = 1$ or 2) file, rename or delete the current file.

If the ADEPT-WFA is upgraded and it is necessary to change the configuration file, the current file will be renamed with an extension of `.save.versionNumber`, where `versionNumber` is the current version of the default version of `wifi_config_te<x>.tcl`. The fields in the `wifi_config_te<x>.tcl` configuration file that can be modified are shown in bold text in [Figure 1-10](#).

```
# WiFi NAV Test Parameters
set WiFi_NAV(11a,chan) 36
set WiFi_NAV(11b,chan) 11
set WiFi_NAV(11g,chan) 11
# WiFi PLCP Test Parameters
set WiFi_PLCP(11a,chan) 36
set WiFi_PLCP(11b,chan) 11
set WiFi_PLCP(11g,chan) 11
# MIC Countermeasure Parameters
set WiFi_MIC(11a,chan) 36
set WiFi_MIC(11b,chan) 11
set WiFi_MIC(11g,chan) 11
set WiFi_MIC(WPA,auth) WPA-PSK
set WiFi_MIC(WPA-PSK,psk) "12345678"
set WiFi_MIC(WPA-PSK,ssid) "wifi"
set WiFi_MIC(WPA2,auth) WPA2-PSK
set WiFi_MIC(WPA2-PSK,psk) "12345678"
set WiFi_MIC(WPA2-PSK,ssid) "wifi"
```

Figure 1-10. `wifi_config_te_<x>.tcl` File Parameters

Obtaining Properties of the ADEPT-WFA Test Engine

All pertinent information about each ADEPT-WFA Test Engine such as version and manufacturing information can be obtained in the ADEPT-WFA Compliance Tester. The properties of each of the ADEPT-WFA's test engines can be obtained by right-clicking the Test Engine and selecting Properties from the menu that displays ([Figure 1-11](#)).

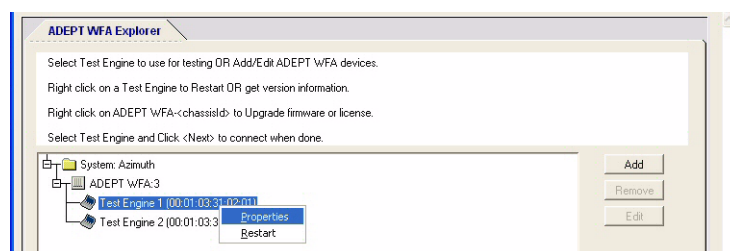


Figure 1-11. Properties Option

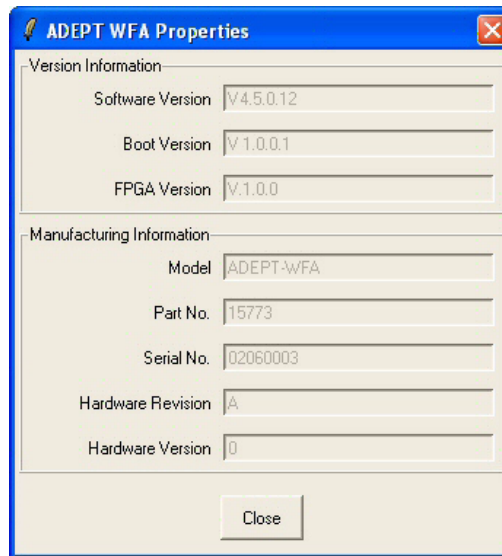


Figure 1-12. Properties Screen Example

Viewing and Configuring Logs

There are three types of log tabs (Figure 1-13) located at the bottom of all of the ADEPT-WFA Compliance Tester screen that you can use to view activities and WFA test results:

- **Log Tab** — log messages concerning events that take place while the Compliance Tester and ADEPT-WFA are communicating display on this tab. This log contains messages about connecting to devices, upgrading firmware, downloading software to devices, rebooting devices, and pinging devices. This tab is saved in the file *output.log*.
- **System Log Tab** — log messages pertaining to the ADEPT-WFA Compliance Tester. These messages typically contain log messages of events that take place on a test engine, including date and time stamps and a description. These messages log such events as crashes, connection failures, MIC Countermeasure status, calibration, and initialization. This tab is saved in the file *system.log*.
- **FTP Log Tab** — event messages of FTP sessions between the ADEPT-WFA Compliance Tester and the test engines. This log contains information such as IP addresses, user names, and passwords that are used between the Compliance Tester and the devices as well as activities performed during those FTP sessions. This tab is saved in the file *ftp.log*.

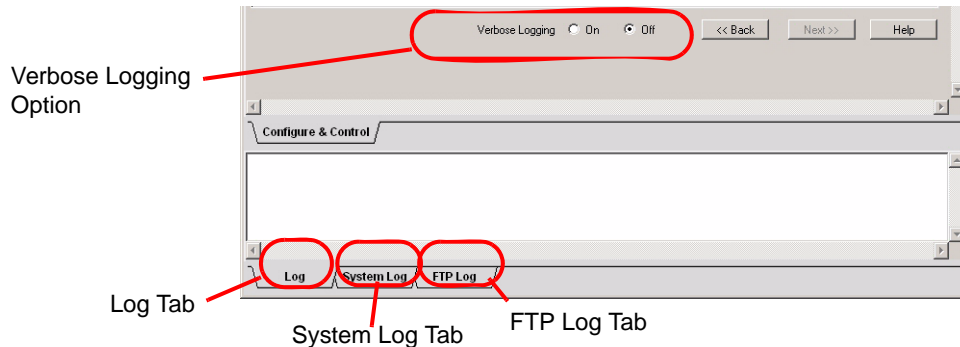


Figure 1-13. Log Tabs

Turning Verbose Logging on or off enables/disables your ability to see debug messages in each of the logs. This option greatly increases the size of each of the log files.

Log files are not deleted when system installations or upgrades are performed. These files can only be deleted manually by the user.

Log files for each *session* on a particular day are saved on the control PC. A session is the time starting when the ADEPT-WFA Compliance Tester is opened and finishing when it is closed. If you have two sessions on a particular date, you will have two subfolders under that date containing each of the log files. All log files are saved in the dated folders under the following path on the control PC (where *date* and *timestamp* are the date and time that the logs were written):

```
Program Files\Azimuth\ADEPT-WFA\data\tests\runDB\ADEPT-
WFA\date\timestamp
```

WiFi Compliance Tests

Tests covered in this section include the Network Allocation Vector (NAV), Physical Layer Convergence Protocol (PLCP), and Message Integrity Check (MIC) tests.

Configuring and Running a NAV Test

The NAV test tab provides the ability to configure and perform the WFA's Network Allocation Vector (NAV) test on a station or AP. The NAV test determines if a device under test (an AP or station) is honoring the NAV field in a Wi-Fi packet.

The NAV is updated with the duration of a received packet(s). During this time the device under test should be treating the medium as busy and not attempt to transmit during that time.

The WFA test plan requires a specified throughput (depending on the radio band) between the station and the AP. During the NAV test, the ADEPT-WFA sends a packet with a specified NAV field setting. If the device under test is honoring the NAV field setting, the throughput must be less than 80% of the throughput when the NAV test is not running.

Next is a summary of what should occur during the NAV test, and an explanation of how the device under test passes/fails the test.

- The WFA test plan's *Data Transfer 1* test is performed and the throughput is recorded. According to the WFA test plan, this throughput must meet a specified throughput (based on its radio band) for the device under test to pass.
- The ADEPT-WFA NAV test is initialized. The operating band, channel and other parameters are set up during initialization.
- The AP and station are configured according to the WFA test plan.
- The station associates with the AP.
- The ADEPT-WFA NAV test is started.
- The ADEPT-WFA sends a CTS-to-self with a large duration field.
- The Chariot script file FILESENDL specified in the WFA test plan for *Data Transfer 1* is sent from the device under test to either the station or AP:
 - For APUT — the transfer should go from the APUT to the station.
 - For STAUT — the transfer must go from the STAUT to the AP.
- Using Chariot, the throughput of the file transfer is observed and recorded.
- Click [Stop Test] to stop the ADEPT-WFA from transmitting CTS-to-self packets.
- If the throughput during the NAV test is less than 80% of the *Data Transfer 1* throughput, the device under test passes.

The following procedure assumes that you have properly installed and powered on the ADEPT-WFA, that the LEDs on the device are in normal operation mode, that the test network has been physically configured, that the ADEPT-WFA Compliance Tester has been installed, that the chassis ID has been assigned to the ADEPT-WFA, and that the IP address and MAC address of the test engines have been assigned.

Configuring and running a NAV test based on the NAV test defined in the WFA's test plan:

1. Configure the AP, station, and test network according to the WFA's test plan (see [Figure 1-14](#) for an example graphic).

Note: Ensure that the wireless devices for this test are at least two feet (802.11a) or five feet (802.11b/g) to the ADEPT-WFA.

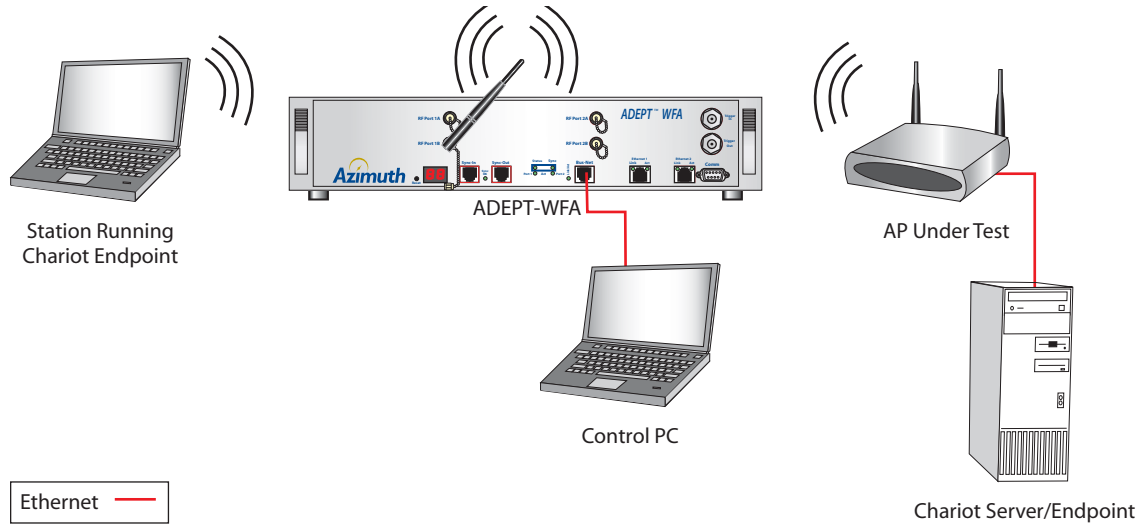


Figure 1-14. STA/AP NAV Tests, Over-the-Air Connection

2. Start the ADEPT-WFA Compliance Tester, select the appropriate test engine, and click [Next>>]. The Tests tab displays.
3. Select the NAV tab (Figure 1-15).

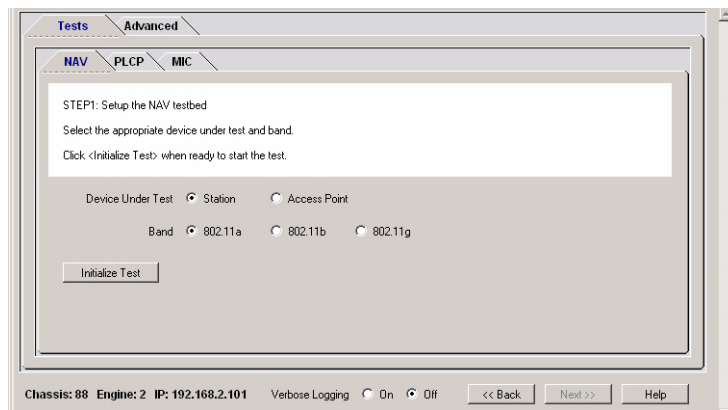


Figure 1-15. NAV Tab

4. Select a Device Under Test radio button (**Station** or **Access Point**).
5. Select a Band radio button (**802.11a**, **802.11b**, or **802.11g**).
6. Click [Initialize Test]. The NAV Test Initialization screen displays (Figure 1-16).

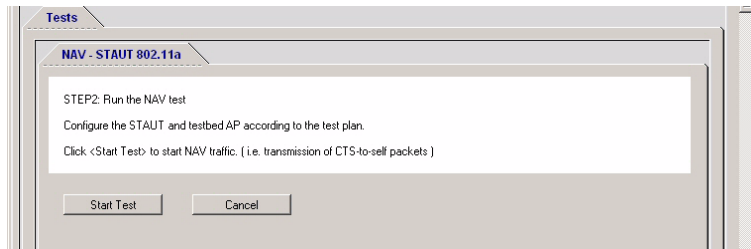


Figure 1-16. NAV Test Initialization Window

7. Click [Start Test] to run the NAV test. The ADEPT-WFA sends a CTS-to-self with a long duration field in a short packet. The elapsed time since the initial transmission of the ADEPT-WFA NAV test packets is shown on the screen.

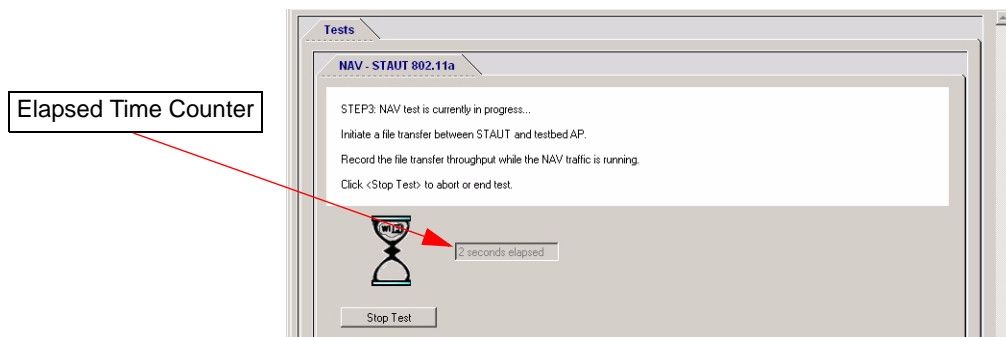


Figure 1-17. NAV Test Status Window

8. Initiate the data transfer defined in the WFA's test plan and use Chariot to check and record the throughput of the file transfer while NAV traffic is running.
9. Click [Stop Test] to stop the NAV test.
10. If the throughput during the NAV test is less than 80%, as specified in the WFA test plan, the device under test passes.

Configuring and Running a PLCP Test

The PLCP test tab enables you to configure and perform the WFA's Physical Layer Convergence Protocol (PLCP) test on a station or AP. The PLCP test determines if a device under test (an AP or station) is honoring the PLCP header length field in a Wi-Fi packet.

The PLCP header length field specifies a packet size for a given 802.11 frame. All devices receiving a PLCP header must mark the channel as busy for the indicated duration. During this time the device under test should be treating the medium as busy to allow packet data transfer.

Next is a summary of what should occur during the PLCP test and an explanation of how the device under test passes/fails the test.

- The WFA test plan's *Data Transfer 1* test is performed and the throughput is recorded. According to the WFA test plan, this throughput must meet a specified throughput (based on its radio band) for the device under test to pass.
- The ADEPT-WFA PLCP test is initialized. The operating band, channel and other parameters are set up during initialization.
- The AP and station are configured according to the WFA test plan.
- The station associates with the AP.
- The ADEPT-WFA PLCP test is started.
- The ADEPT-WFA sends a test frame with a PLCP length field that is much larger than the actual number of bytes being transmitted.
- The Chariot script file FILESENDL specified in the WFA test plan for *Data Transfer 1* is sent from the device under test to either the station or AP:
 - For APUT — the transfer should go from the APUT to the station.
 - For STAUT — the transfer must go from the STAUT to the AP.
- Using Chariot, the throughput of the file transfer is observed and recorded.
- Click [Stop Test] to stop the PLCP test.
- If the throughput during the PLCP test is less than 80% of the *Data Transfer 1* throughput, the device under test passes.

The following procedure assumes that you have properly installed and powered on the ADEPT-WFA, that the LEDs on the device are in the normal operational mode, that the test network has been physically configured, that the ADEPT-WFA Compliance Tester has been installed, that the chassis ID has been assigned to the ADEPT-WFA, and that the IP address and MAC address of the test engines have been assigned.

Configuring and running PLCP Tests based on the PLCP test defined in the WFA's test plan:

1. Configure the AP, station, and test network according to the WFA's test plan (see [Figure 1-18](#) for an example graphic).

Note: Ensure that the wireless devices for this test are at least two feet (802.11a) or five feet (802.11b/g) to the ADEPT-WFA.

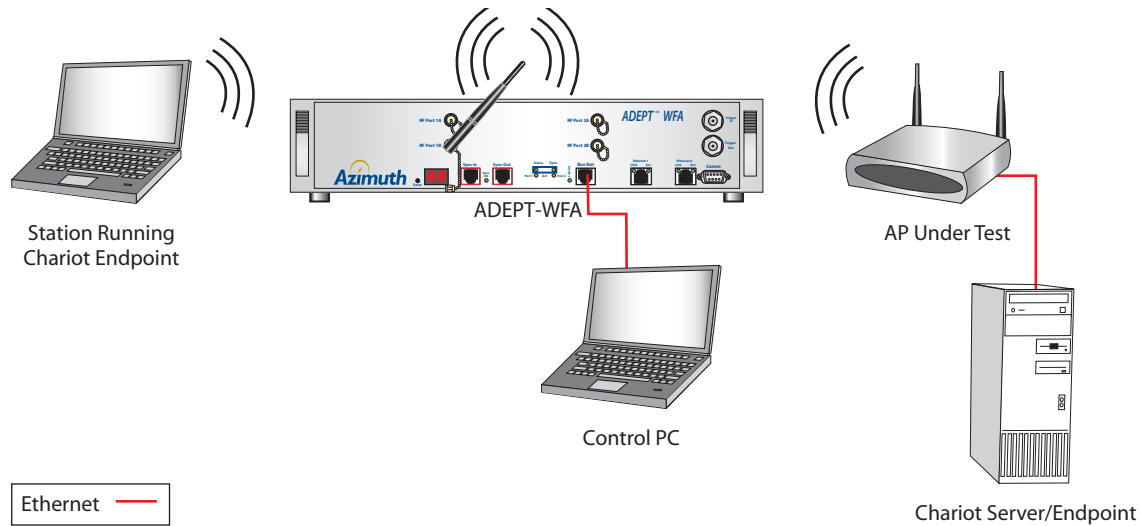


Figure 1-18. STA/AP PLCP Tests, Over-the-Air Connection

2. Start the ADEPT-WFA Compliance Tester, select the appropriate test engine, and click [Next>>]. The Tests tab displays.
3. Select the PLCP tab (Figure 1-15).

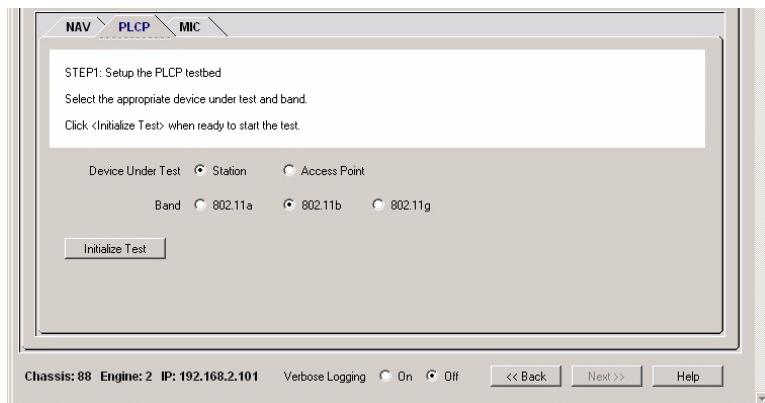


Figure 1-19. PLCP Tab

4. Select a Device Under Test radio button (**Station** or **Access Point**).
5. Select a Band radio button (**802.11a**, **802.11b**, or **802.11g**).
6. Click [Initialize Test]. The PLCP Test Initialization screen displays (Figure 1-16).

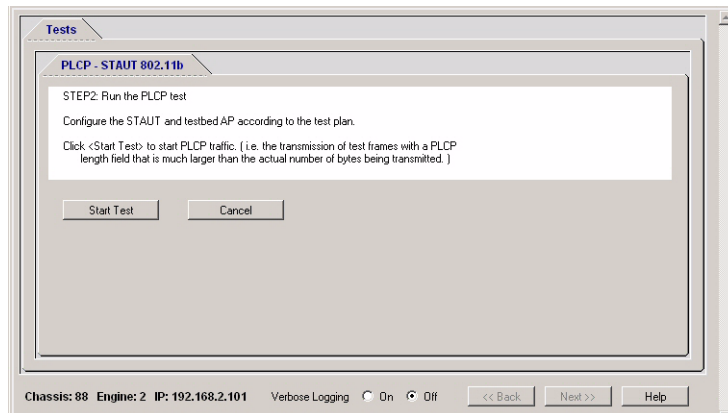


Figure 1-20. PLCP Test Initialization Window

7. Click [Start Test] to run the PLCP test. The ADEPT-WFA sends a test frame with a PLCP length field that is much larger than the actual number of bytes being transmitted. The elapsed time since the initial transmission of the ADEPT-WFA NAV test packets is shown on the screen.

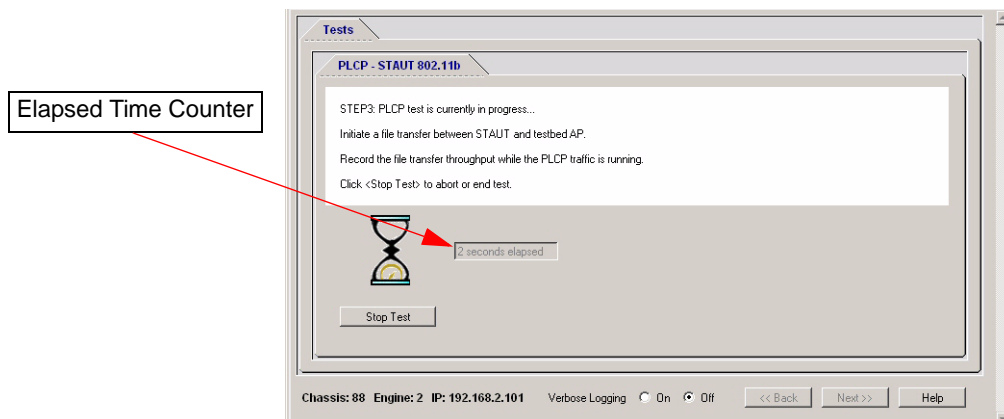


Figure 1-21. PLCP Test Status Window

8. Initiate the data transfer defined in the WFA’s test plan and use Chariot to check and record the throughput of the file transfer while PLCP traffic is running.
9. Click [Stop Test] to stop the PLCP test.
10. If the throughput during the PLCP test is less than 80%, as specified in the WFA test plan, the device under test passes.

Configuring and Running a MIC Test

The MIC test tab enables you to configure and perform the WFA’s Message Integrity Check (MIC) Countermeasures test on a station or AP. The MIC test determines if a device under test behaves appropriately when it receives two packets containing a corrupt MIC field in less than 60 seconds.

Configuring/Running the STAUT MIC Test

The MIC test for a STAUT determines if the station responds appropriately when it receives packets containing a corrupt MIC field, and that the STAUT behaves appropriately when two packets containing a corrupt MIC field are received in less than one minute. The procedure in this section describes how to configure and run the MIC test for a STAUT.

Next is a summary of what should occur during the STAUT MIC test and an explanation of how the station passes/fails the test. Remember that in this test case, the ADEPT-WFA emulates an AP.

- The ADEPT-WFA MIC Countermeasures test is initialized. The operating band, channel and other parameters are set up during initialization.
- The STAUT associates with the ADEPT AP.
- Start a ping session for the STAUT to the ADEPT AP.
- For WPA2 tests, start the broadcast ping from the PC connected to the **Ethernet 1** port, which provides packets that the ADEPT-WFA will use to corrupt MIC fields.
- The ADEPT AP sends one echo reply packet containing a corrupt MIC field.
- The STAUT sends a first MIC failure report to the AP and continues its ping session.
- Seventy seconds later, the AP sends a second echo reply packet containing a corrupt MIC field.
- The STAUT sends a second MIC failure report to the AP and continues its ping session.
- Fifty seconds later, the AP sends a third echo reply packet containing a corrupt MIC field.
- Upon reception of the third echo reply, the STAUT sends a third MIC failure report to the AP. The STAUT must then deauthenticate and disassociate itself from the ADEPT AP. Verify that pings from the STAUT are interrupted, indicating that the STAUT is no longer connected to the ADEPT AP.
- One minute later, the STAUT reassociates with the AP and the ping session is re-established.

If the expected behavior of the STAUT is not met, the test fails. To pass the MIC test, the STAUT must send all MIC failure reports at the expected times, deauthenticate and disassociate after receiving the third packet containing the corrupt MIC field, and reassociate with the AP at or after 60 seconds from disassociating/deauthenticating.

Note that passing the test requires the STAUT to disassociate/deauthenticate only when two corrupt packets are received in less than sixty seconds. Since the second packet is sent seventy seconds after the first, the STAUT should not disassociate/deauthenticate when the second packet is received. The third packet is sent less than a minute after the second packet, so the STAUT should disassociate/deauthenticate upon reception of the third packet.

The following procedure assumes that you have properly installed and powered on the ADEPT-WFA, that the LEDs on the device are in the normal operational mode, the ADEPT-WFA Compliance Tester has been installed, the chassis ID has been assigned to the ADEPT-WFA, and the IP address and MAC address of Test Engine 1 has been assigned.

Configuring and running the STAUT MIC Test based on the STAUT MIC test defined in the WFA's test plan:

1. Configure the station and test network according to the WFA's test plan (see [Figure 1-22](#) for an example graphic).

Note: Ensure that the wireless devices for this test are at least two feet (802.11a) or five feet (802.11b/g) to the ADEPT-WFA.

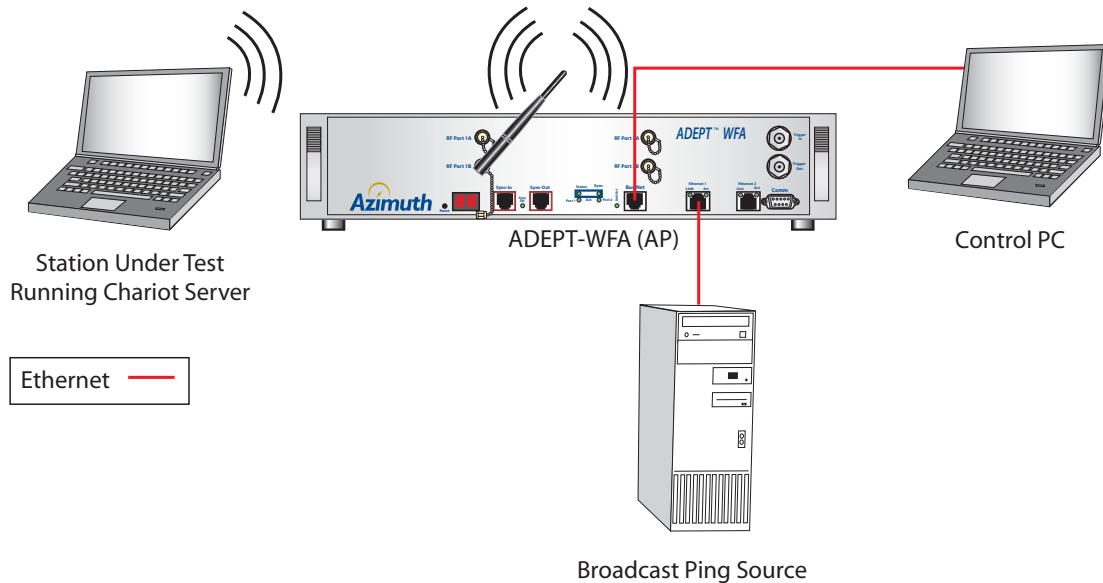


Figure 1-22. STAUT MIC Test, Over-the-Air Connection

2. Start the ADEPT-WFA Compliance Tester, select the appropriate test engine, and click [Next>>]. The Tests tab displays.
3. Select the MIC tab ([Figure 1-23](#)).

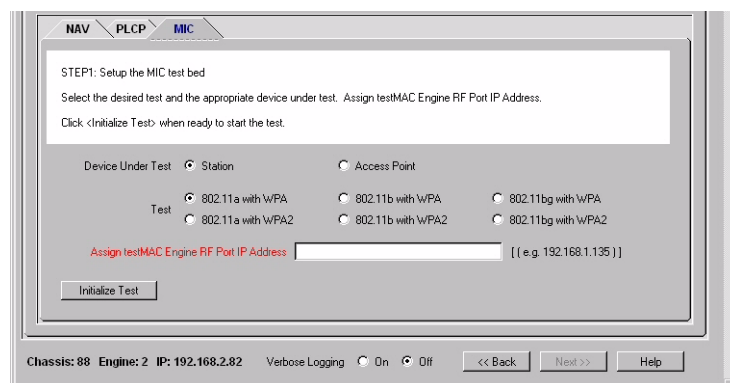


Figure 1-23. MIC Tab

4. Select the **Station** radio button as the Device Under Test.
5. Select a Test radio button (e.g., 802.11a with WPA2).

6. Enter the IP address of the ADEPT-WFA in the **Assign testMAC Engine RF Port IP Address** field. (This is the destination address for the ping session from the STAUT session.) This IP address must be on a different subnet than the Bus-Net IP address.
7. Click [Initialize Test]. The screen for setting up the STAUT displays (Figure 1-24).

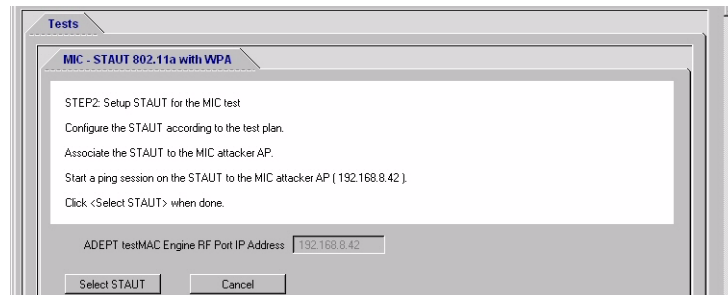


Figure 1-24. Setting Up the STAUT Window

Note: Be aware of the direction given on the MIC test screen.

8. Click [Select STAUT]. The screen for running the MIC test displays (Figure 1-25).

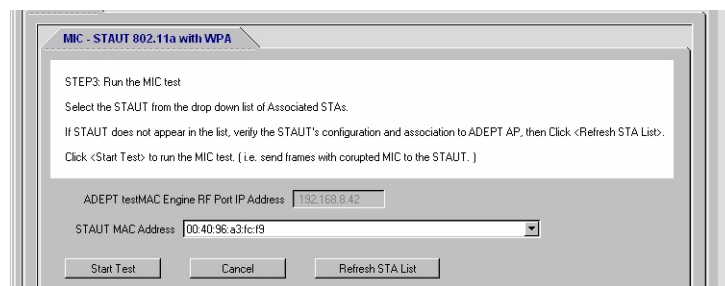


Figure 1-25. Run the MIC Test Window

9. Select the MAC address of the STAUT from among the options in the **STAUT MAC Address** pull-down menu. (This information appears after the STAUT associates with the ADEPT-WFA AP.)

If the desired STAUT MAC address does not appear, select [Refresh STA List] so that the ADEPT-WFA updates the list of associated STAs; check the drop-down menu to select the desired STAUT.

When testing WPA2, provide a broadcast ping from the PC connected to the **Ethernet 1** port to the same subnet as the ADEPT Test Engine RF Port IP address.

10. Click [Start Test]. The first packet with a corrupt MIC field is sent. A rotating hourglass appears in the first status window (Figure 1-26) to indicate that the test is running. The ADEPT-WFA AP expects to receive a MIC Failure Report from the STAUT. If a MIC Failure Report is not received, the test stops and indicates the test failed.

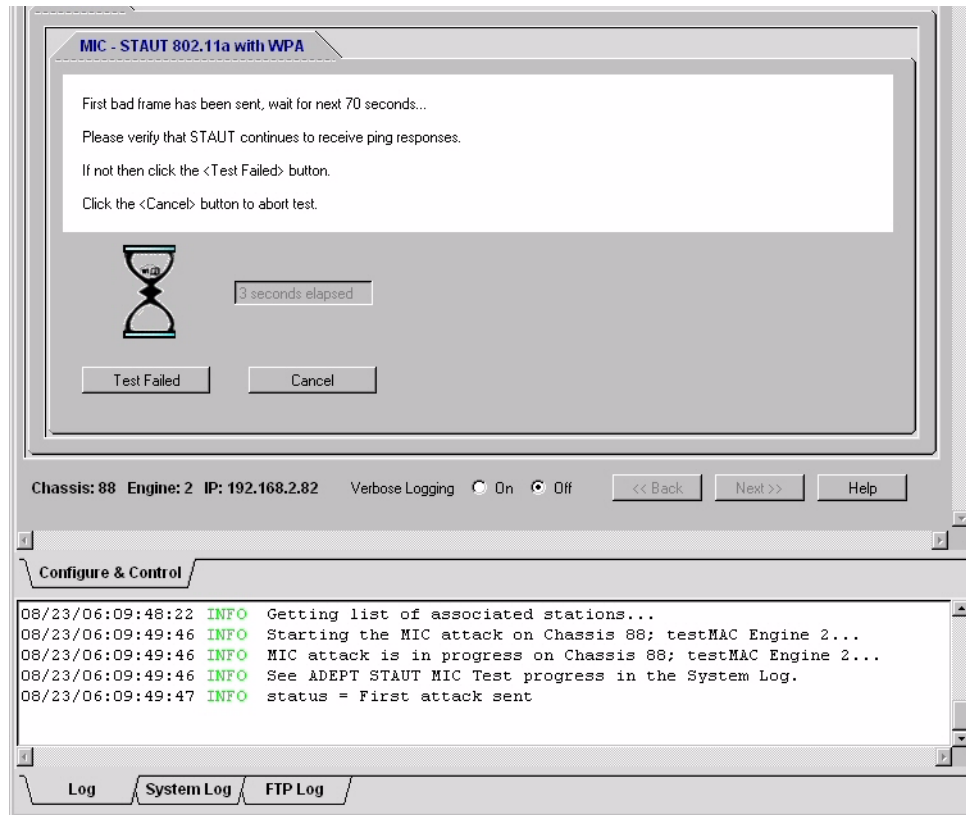


Figure 1-26. First MIC Test Status Window

11. During the next seventy seconds, ensure that the STAUT continues to receive ping responses. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 7](#).
12. Seventy seconds after the AP (the ADEPT-WFA) sends the first packet with a corrupt MIC field, the AP sends a second packet with a corrupt MIC field, which is reflected in the status message in the window. Note that the elapsed time since sending the first packet is specified in a timer in the window (see [Figure 1-26](#)). The ADEPT-WFA AP expects to receive a MIC Failure Report from the STAUT. If a MIC Failure Report is not received, the test stops and indicates the test failed.
13. During the next fifty seconds, ensure that the STAUT continues to receive ping responses. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 7](#).
14. Fifty seconds later, the AP sends the third packet containing the corrupt MIC field. If a MIC Failure Report is not received, the test stops and indicates the test failed. The ADEPT-WFA AP expects to receive a MIC Failure Report from the STAUT. If a MIC Failure Report is not received, the test stops and indicates the test failed.

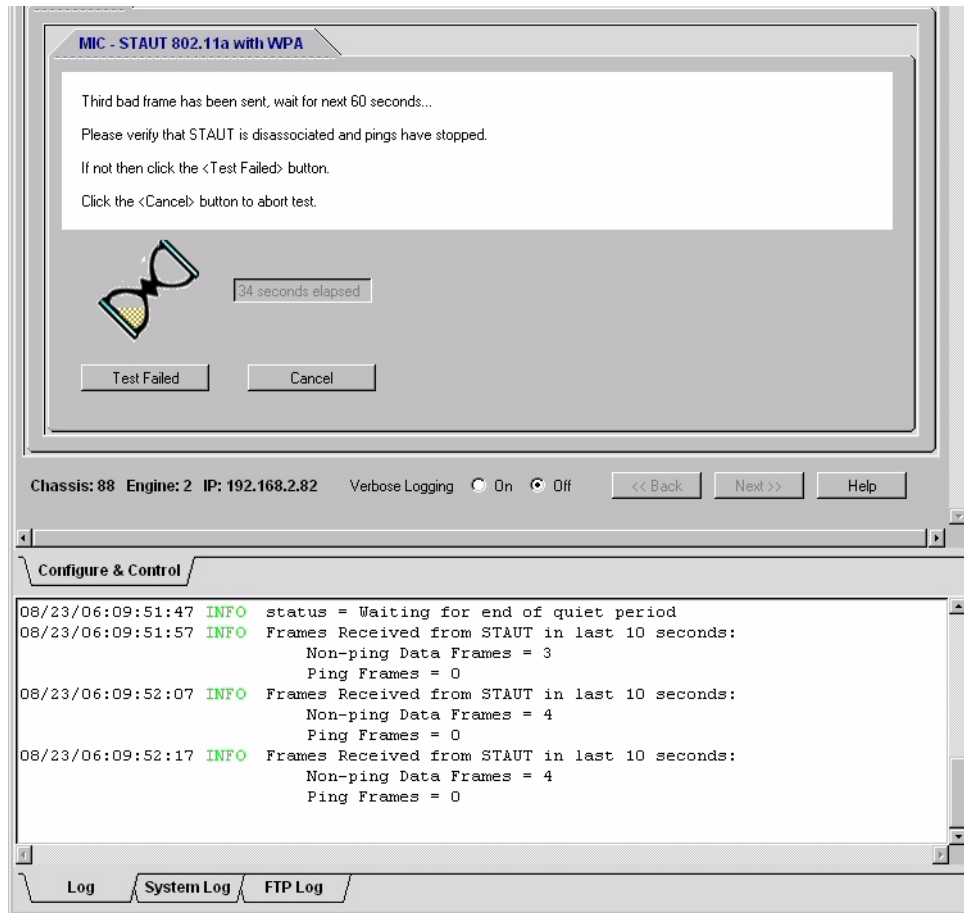


Figure 1-27. Third MIC Test Status Window

15. Ensure that the STAUT has deauthenticated/disassociated from the AP and that all pings have stopped. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 7](#).

The ADEPT-WFA firmware does not automatically fail a STAUT for ping or data frames received from the STAUT during the quiet period. For example, EAPOL data frames would be counted in the dataPkts. The pingPkts value contains the number of ICMP data frames received by the ADEPT AP (destination MAC address) and sent by the STAUT (source MAC address). Note that these counter values return the total number of frames received after the third MIC Failure Report until the STAUT reassociates to the ADEPT AP.

The ADEPT-WFA Compliance Tester retrieves the MIC counters at 10 second intervals beginning at the start of the quiet period until the STAUT reassociates to the ADEPT AP. This gives details of when these packets are transmitted. At the very least, these counters must be checked at test completion to get the total number of data and ping frames received during the quiet period.

Ideally, an AP should not receive any data frames from a STA that is not associated to it; non-ping data frames that are not actually attempting to forward data to another station have been allowed, but the ping packets (attempting to forward data to an end station), in general, are not allowed. If the pingPkt value is not zero, the test should be considered a failure.

16. The ADEPT-WFA waits for sixty seconds. If the STAUT attempts to re-associate during that time, the test will fail.
17. After the sixty second quiet period, the ADEPT-WFA AP expects the STAUT to re-associate to it and resume pinging. If the STAUT fails to re-associate and transmit ping requests, the test will fail.
18. A message appears to indicate if the MIC STAUT test has passed or failed, and (if applicable) includes some type of brief explanation of when the test failed.
Examples of pass/fail status messages that may appear during this test include the following (see the complete list of status messages in [Appendix A, “ADEPT-WFA Compliance Tester Tcl Commands”](#) (page A-1) and [Appendix B, “ADEPT-WFA Tcl Commands”](#) (page B-1):
 - ❑ MIC STAUT Status: PASSED
 - ❑ MIC STAUT Failed by User: Ready for first attack
 - ❑ MIC STAUT Aborted by User: Ready for first attack
 - ❑ WPA STAUT Failed: Not able to send second MIC attack. Please verify STAUT is continuing to receive ping responses.

Configuring/Running the APUT MIC Test

The MIC test for an APUT determines if the AP responds appropriately when the APUT receives packets containing a corrupt MIC field or MIC failure reports, and that the APUT behaves appropriately when two corrupt packets or MIC failure reports are received in less than one minute. The APUT must not allow the test stations to re-associate before the sixty second mark from deauthenticating/disassociating the stations. The procedure in this section describes how to configure and run the MIC test for an APUT.

Next is a summary of what should occur during the APUT MIC test and an explanation of how the AP passes/fails the MIC test. Remember that in this test case, the ADEPT-WFA emulates a test station.

- The ADEPT-WFA MIC Countermeasures test is initialized. The operating band, channel and other parameters are set up during initialization.
- The ADEPT-WFA test station and the two other test bed stations associate with the APUT.
- The two test bed stations each send an echo request (ping) to a ping endpoint.
- The ADEPT-WFA station sends an echo request packet containing a corrupt MIC field to the ping endpoint.
- Seventy seconds later, the ADEPT-WFA station sends a second echo request packet containing a corrupt MIC field to the ping endpoint.
- Fifty seconds later, the ADEPT-WFA station sends a third echo request packet containing a corrupt MIC field to the ping endpoint.
- Upon reception of the third echo request packet containing the corrupt MIC field, the APUT deauthenticates and disassociates all associated stations.
- The ADEPT-WFA station attempts to associate and send pings to verify that the APUT will not forward traffic at this time.
- One minute later, the APUT allows all of the test stations to reassociate to it.

- The ADEPT-WFA test station and the two other test bed stations associate with the APUT.
- The two test bed stations each send an echo request (ping) to a ping endpoint.
- The ADEPT-WFA station sends a first MIC failure report to the ping endpoint.
- Seventy seconds later, the ADEPT-WFA station sends a second MIC failure report to the ping endpoint.
- Fifty seconds later, the ADEPT-WFA station sends a third MIC failure report to the ping endpoint.
- Upon reception of the third MIC failure report, the APUT deauthenticates and disassociates all associated stations.
- The ADEPT-WFA station attempts to associate and send pings to verify that the APUT will not forward traffic at this time.
- One minute later, the APUT allows all of the test stations to reassociate with it.

If the expected behavior of the APUT is not met, the test fails. To pass the MIC test, the APUT must perform the following:

- Deauthenticate and disassociate stations (ping traffic stops) after receiving the third packet containing the corrupt MIC field, not allow the ADEPT-WFA station to associate and resume the ping session, and allow reassociation with the test bed stations sixty seconds later.
- Deauthenticate and disassociate after receiving the third MIC failure report, not allow the ADEPT-WFA station to associate and resume the ping session, and allow reassociation with the test bed stations sixty seconds later.

Note that passing the test requires the APUT to disassociate/deauthenticate only when two corrupt packets or MIC failure reports are received in less than sixty seconds. Since the second corrupt packet and MIC failure report is sent seventy seconds after the first, the APUT should not disassociate/deauthenticate when the second packet or report is received. The third packet or report is sent less than a minute after the second packet, so the APUT should disassociate/deauthenticate upon reception of the third packet or report.

The following procedure assumes that you have properly installed and powered on the ADEPT-WFA, that the LEDs on the device are in the normal operational mode, the ADEPT-WFA Compliance Tester has been installed, the chassis ID has been assigned to the ADEPT-WFA, and the IP address and MAC address of Test Engine 1 has been assigned.

Configuring and running the APUT MIC Test based on the APUT MIC test defined in the WFA's test plan:

1. Configure the AP and test network according to the WFA's test plan (see [Figure 1-28](#) for an example graphic).

Note: Ensure that the wireless devices for this test are at least two feet (802.11a) or five feet (802.11b/g) to the ADEPT-WFA.

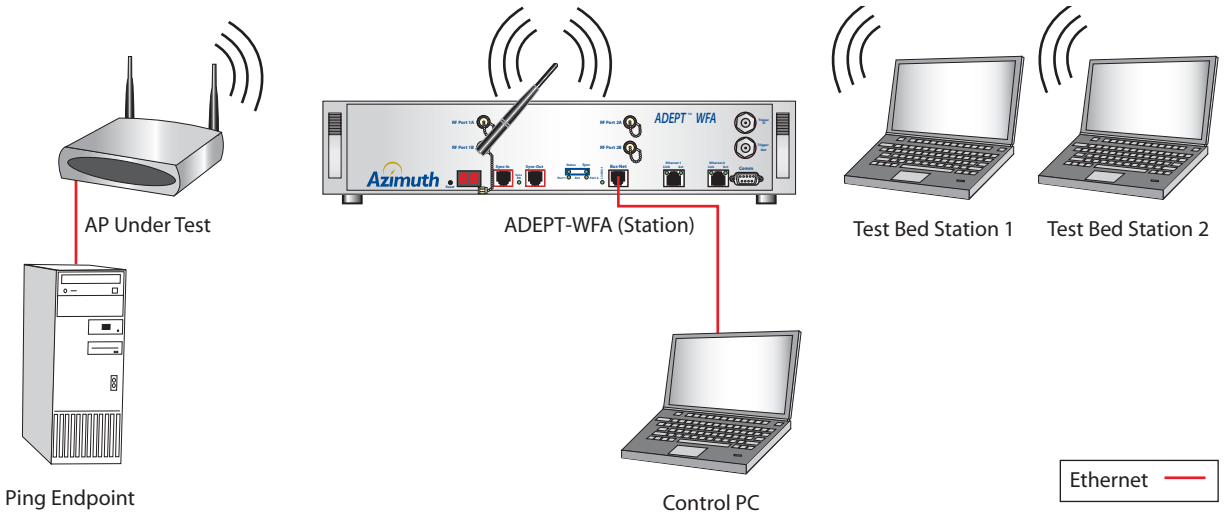


Figure 1-28. APUT MIC Test, Over-the-Air Connection

2. Start the ADEPT-WFA Compliance Tester, select the appropriate test engine, and click [Next>>]. The Tests tab displays
3. Select the MIC tab (Figure 1-29).

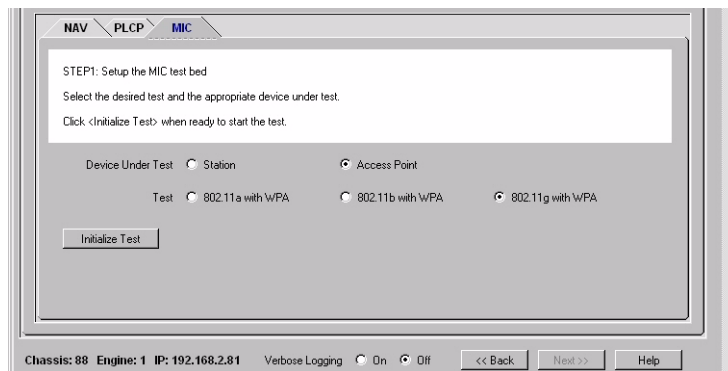


Figure 1-29. MIC Tab

4. Select the **Access Point** radio button as the Device Under Test.
5. Select a Test radio button (**802.11a with WPA, 802.11b WPA, 802.11g WPA**).
6. Click [Initialize Test]. The screen for setting up the APUT displays (see Figure 1-30).

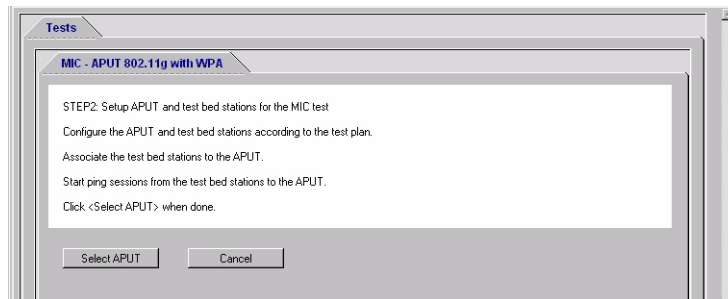


Figure 1-30. Setting Up the APUT Window

Note: Be aware of the direction given on the MIC test screen.

- Click [Select APUT]. The configuration screen displays (Figure 1-31).

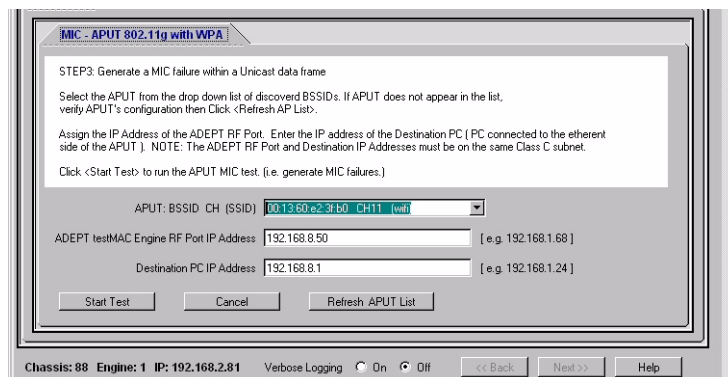


Figure 1-31. MIC Configuration Window

- Select the BSSID of the APUT from the pull-down menu in the **APUT BSSID CH (SSID)** field. (This information appears after the ADEPT-WFA station performs a scan on the selected wireless band.)

If the desired APUT MAC address does not appear, select [Refresh APUT List] so that the ADEPT-WFA will rescan for the desired APUT. Then check the pull-down menu again to select the desired APUT.

- Enter the IP address of Test Engine 1's RF (Wi-Fi) port in the **ADEPT Test Engine RF Port IP Address** field. This IP address must be on a different subnet than the Bus-Net IP address and must be on the same Class C subnet as the **Destination PC IP Address** field.
- Enter the IP address of the ping endpoint PC connected to the Ethernet port (Distributed Services (DS) interface) of the APUT in the **Destination PC IP Address** field. This IP address must be on a different subnet than the Bus-Net IP address and must be on the same Class C subnet as the **ADEPT Test Engine RF Port IP Address** field.
- Click [Start Test]. The ADEPT-WFA station sends the first packet with a corrupt MIC field to the ping endpoint (Figure 1-32).

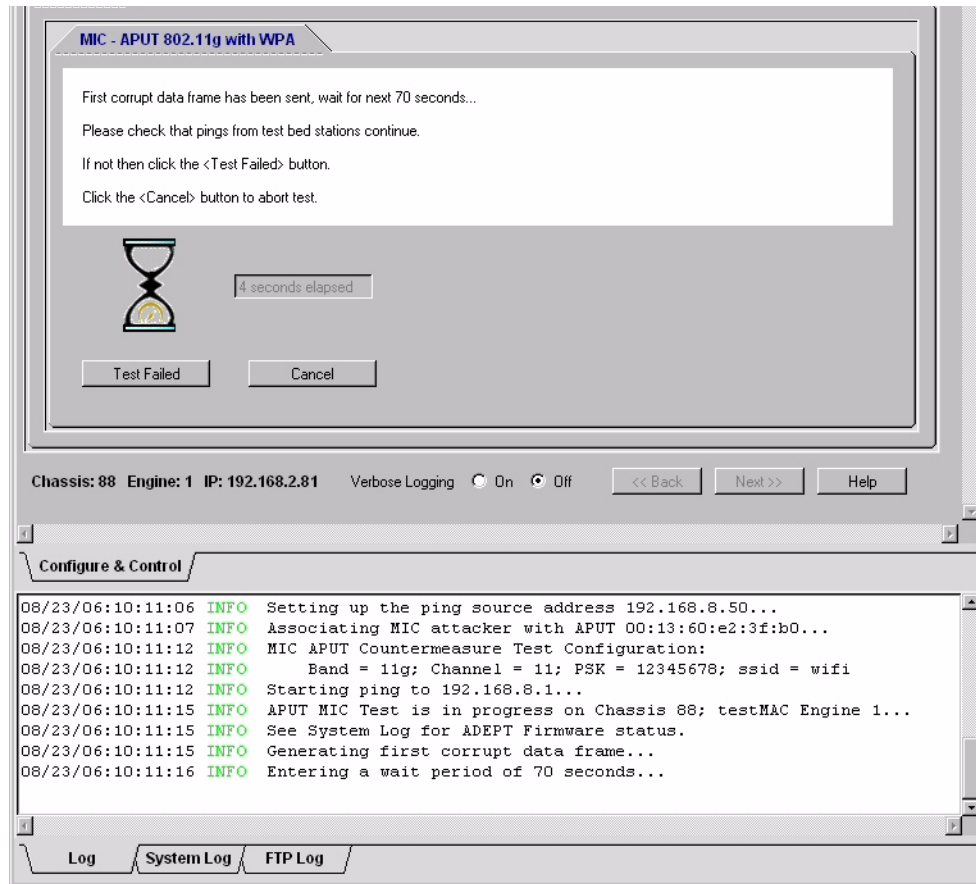


Figure 1-32. First MIC Test Status Window

The ADEPT-WFA station automatically starts a ping session to the ping endpoint through the APUT and sends the first packet with a corrupted MIC field.

12. Ensure that all pings from the two test bed stations continue. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).

Seventy seconds after the ADEPT-WFA station sends the first packet, the ADEPT-WFA station sends a second packet with a corrupt MIC field, which is reflected in the status message in the window. Note that the elapsed time since sending the first packet is specified in a timer in the window (see [Figure 1-32](#)).

13. Continue to ensure that all pings from the two test bed stations continue. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).
14. Fifty seconds later, the ADEPT-WFA station sends the third packet containing the corrupt MIC field, which is reflected in the status message in the window. The Compliance Tester starts a 60 second timer after the third packet containing the corrupt MIC field is sent. During this time, the ADEPT-WFA station attempts to associate to the APUT. If the APUT allows the ADEPT-WFA station to associate prior to the 60 second timer ending, the test fails. Ensure that the APUT deauthenticates/disassociates all three test stations. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).

15. Prepare for the second half of the test - MIC failure report: A Test bed Station screen displays prompting you to take the following actions:
 - a. **Associate the test bed station(s) to the APUT.** The ADEPT-WFA station automatically associates with the APUT. The APUT should not allow this until sixty seconds after deauthentication/disassociation.
 - b. **Start a ping session from the test bed station(s) to the APUT.** The ADEPT-WFA station automatically starts a ping session to the ping endpoint through the APUT.
16. After taking the actions described in [step 14](#), click [OK] to close the prompt dialog box. The ADEPT-WFA station sends the first MIC failure report to the ping endpoint.
17. Ensure that all pings from the two test bed stations continue. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).
18. Seventy seconds after the ADEPT-WFA station sends the first MIC failure report, the ADEPT-WFA station sends a second MIC failure report to the ping endpoint, which is reflected in the status message in the window.
19. Continue to ensure that all pings from the two test bed stations continue. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).
20. Fifty seconds later, the ADEPT-WFA station sends the third MIC failure report to the ping endpoint, which is reflected in the status message in the window. The Compliance Tester starts a 60 second timer after the third packet containing the corrupt MIC field is sent. During this time, the ADEPT-WFA station attempts to associate to the APUT. If the APUT allows the ADEPT-WFA station to associate prior to the 60 second timer ending, the test fails. Ensure that the APUT deauthenticates/disassociates all three test stations. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).
21. Ensure that all pings from the two test bed stations continue. If not, click [Test Failed], make any necessary adjustments and repeat the test from [step 6](#).
22. A message appears to indicate if the MIC APUT test has passed or failed, and (if applicable) includes some type of brief explanation of when the test failed.

Examples of pass/fail status messages that may appear during this test include the following:

- MIC APUT Status: PASSED
- MIC APUT Failed by User: After first corrupt data frame during verify pings and 4 seconds elapsed
- MIC APUT Aborted by User: After first corrupt data frame during verify pings and 11 seconds elapsed
- MIC APUT Failed: No ping response received. After first corrupt data frame and 21 seconds elapsed.

Appendix A: ADEPT-WFA Compliance Tester Tcl Commands

This chapter describes how to create Tcl scripts to control the ADEPT-WFA to run the NAV, PLCP, and MIC Countermeasure tests.

The ADEPT-WFA provides a Tk-based Windows user interface that makes calls to the Tcl commands described in detail in “ADEPT-WFA Tcl Commands” (on page B-1). This chapter explains how to initiate an Azimuth Tcl session and the sequence of commands to call in order to run the three tests.

Controlling the ADEPT-WFA using scripts may be preferred if you are currently running these tests using scripts and/or are controlling other testbed devices using Tcl scripts. For details on the NAV, PLCP, or MIC Countermeasure tests and testbed setups, reference the *Wi-Fi 802.11 with WPA2, WPA, and WEP System Interoperability Test Plan for IEEE 802.11a, b & g Devices* and the following sections in this document:

- ❑ “Configuring and Running a NAV Test” (on page 1-11)
- ❑ “Configuring and Running a PLCP Test” (on page 1-14)
- ❑ “Configuring and Running a MIC Test” (on page 1-17)

This chapter assumes that you have basic knowledge of the Tcl scripting language. For more information on Tcl programming, refer to the book *Practical Programming in Tcl and Tk* by Brent Welch and Ken Jones, with Jeffrey Hobbs, available from Prentice Hall, the TclTutor from <http://www.msen.com/~clif/TclTutor.html>, or one of the vast Tcl Programming resources available on the Internet.

Using Tcl scripts to control the ADEPT-WFA hardware requires a PC (minimum requirements described in *Azimuth ADEPT-WFA Installation and Upgrade Guide* (pub. no. 15849) with the Azimuth ADEPT-WFA Compliance Tester application installed. The installation includes a Tcl shell (version 8.4) and required Azimuth Systems library files.

Start by initiating an interactive Tcl session, then enter Tcl commands, or call a programmed script. These steps are detailed in the following sections.

The ADEPT-WFA utilizes Azimuth’s testMac Module (TMM) technology, so many of the Tcl commands begin with the prefix *tmm_*.

Running any of the ADEPT-WFA tests consists of three major steps:

- ❑ Configure the ADEPT-WFA hardware.
- ❑ Start the test.
- ❑ Stop the test.

Depending on the differences in setup between the test you are about to run and the test previously run, the ADEPT-WFA hardware might require a reboot. If required, the ADEPT-WFA configuration Tcl commands automatically reboot the hardware. The configuration tcl commands are blocking a return to the control prompt until reboot and configuration are complete; if necessary, wait until the testMAC Engine completes rebooting and the standalone PC reconnects to the testMAC engine.

Note that only one test can be run at any given time on a testMAC Engine. Attempting to simultaneously run NAV traffic, PLCP traffic or a MIC Countermeasure test on a testMAC Engine will yield unpredictable results.

Many of the Azimuth Tcl procedures will return using the 'error' Tcl command if it cannot complete successfully. It is recommended to wrap the Tcl 'catch' command around calls to the ADEPT-WFA Tcl library functions to catch these error conditions. Errors will occur if incorrect parameter values are passed in functions (for example an invalid IP address), or if the ADEPT-WFA hardware is not able to complete the function successfully. For example, if the *tmm_connect_ap* procedure is called and the MAC address of the AP to connect to is not heard by the ADEPT-WFA hardware, it will return the message "AP not found."

In addition, the Azimuth SDK includes support for the *\$errorInfo* variable. If an error occurs in the Tcl shell, detailed information about the error can be obtained by issuing the following puts statement immediately after the error occurs:

```
puts $errorInfo
```

Initiating a Tcl Session

To use the ADEPT-WFA Tcl commands, you must perform the following steps:

1. Perform the quick startup instructions presented in the Quick Start guide and verify the ability to connect to a testMAC Engine via the Compliance Tester. Open Windows Explorer and navigate to the following path:

```
C:\Program Files\Azimuth\ADEPT-WFA\bin\
```

2. Double-click the file *tclsh.exe*. This opens the Tcl shell and the percent (%) sign prompt appears. Enter the following series of commands to initiate a session with the desired testMAC Engine on the ADEPT-WFA (commands are case-sensitive).

```
package require Azimuth-Sdk  
connect_device Azimuth-DEP <testMAC Engine IP Address>
```

When you finish entering these commands the percent (%) sign displays. You are now ready to enter ADEPT-WFA Tcl commands. The *connect_device* command will also set the Time Of Day on the testMAC Engine if an NTP server is not discovered by the ADEPT-WFA. The *connect_device* command will return the target ID that must be used in all other ADEPT-WFA commands and should be stored in a variable. For example:

```
set target connect_device Azimuth-DEP <Test Engine IP address>  
tmm_set_console_ip $target <IP address of host PC Interface to ADEPT-  
WFA>
```

The `tmm_set_console_ip` command sets the IP address of the controlling PC (where Tcl scripts are being run) in the ADEPT-WFA testMAC Engine. When this parameter is set, the testMAC Engine will be able to send System Logs and Output Logs the host PC (refer to the section ‘Viewing and Configuring Logs’). The following sections assume that the return from `connect_device` is ‘Azimuth-DEP-1’. For scripting purposes, it is recommended to use the above example to set the target variable and to use it in the commands (i.e. replace Azimuth-DEP-1 with *\$target*).

Running the NAV Test

Scripting the NAV test requires five simple steps:

- ❑ Configure the ADEPT-WFA hardware.
- ❑ Configure the testbed.
- ❑ Start the ADEPT-WFA NAV traffic.
- ❑ Start the Chariot file transfer.
- ❑ Stop the NAV traffic after the file transfer completes.

Configure ADEPT-WFA for the NAV test by calling the following command:

```
tmm_config_nav_test Azimuth-DEP-1 <band>
```

Replace `<band>` with the desired band to run the test on (either 11a, 11b, or 11g). To run the NAV test on another channel, use the optional parameter, `-channel`. For example, to run the NAV test on channel 52, use the following command:

```
tmm_config_nav_test Azimuth-DEP-1 11a -channel 52
```

Configure the DUT and testbed devices, and then start the ADEPT-WFA NAV traffic with this command:

```
tmm_start_nav_test Azimuth-DEP-1
```

Verifying that the state of NAV traffic on the ADEPT-WFA hardware can be obtained by calling the `get status` command:

```
tmm_get_nav_test_status Azimuth-DEP-1
```

This command will return “Running” (if the NAV traffic is being transmitted) or “Not Running” (if NAV traffic is stopped).

Now run the data transfer test by starting the FILESENDL script in the Chariot testbed PC. When the file transfer has completed, the ADEPT-WFA NAV traffic is stopped using this command:

```
tmm_stop_nav_test Azimuth-DEP-1
```

Running the PLCP Test

Scripting the PLCP test requires five simple steps

- ❑ Configure the ADEPT-WFA hardware.

- ❑ Configure the testbed.
- ❑ Start the ADEPT-WFA PLCP traffic.
- ❑ Start the Chariot file transfer.
- ❑ Stop the PLCP traffic after the file transfer completes.

Configure ADEPT-WFA for the PLCP test by calling the following command:

```
tmm_config_plcp_test Azimuth-DEP-1 <band>
```

Replace <band> with the desired band to run the test on (either 11a, 11b, or 11g). To run the PLCP test on another channel, use the optional parameter, *-channel*. For example, to run the PLCP test on channel 9 on wireless band 11g, use the following command:

```
tmm_config_nav_test Azimuth-DEP-1 11g -channel 9
```

Configure the DUT and testbed devices, and then start the ADEPT-WFA PLCP traffic using this command:

```
tmm_start_plcp_test Azimuth-DEP-1
```

Verify that the state of PLCP traffic on the ADEPT-WFA hardware can be obtained by calling the *get status* command:

```
tmm_get_plcp_test_status Azimuth-DEP-1
```

This command will return: "Running" (if the PLCP traffic is being transmitted) or "Not Running" (if PLCP traffic is stopped).

Now run the data transfer test by starting the FILESENDL script in the Chariot testbed PC. When the file transfer has completed, the ADEPT-WFA PLCP traffic is stopped by issuing this command:

```
tmm_stop_plcp_test Azimuth-DEP-1
```

Running the STAUT MIC Countermeasure Test

Scripting the STAUT MIC test is similar to the NAV and PLCP tests:

- ❑ Configure the ADEPT-WFA hardware.
- ❑ Configure the STAUT.
- ❑ Configure the testbed devices.
- ❑ Start testbed traffic.
- ❑ Start the ADEPT-WFA MIC test.
- ❑ Monitor status and counters for the STAUT MIC test.

Configure ADEPT-WFA using the following command:

```
tmm_config_staut_mic_test Azimuth-DEP-1 <band> <security> <attackerIp>
```

Replace <band> with the desired band to run the test on (either 11a, 11b, or 11bg). Replace <security> with either "WPA-PSK" or "WPA2-PSK". The <attackerIp> parameter is the IP address that will be assigned to the ADEPT-WFA AP's RF Port and is the destination address of the ping session from the STAUT.

Note: The ADEPT-WFA RF address must NOT be on the same subnet as the ADEPT-WFA Bus-Net address (used in the *connect_device* command) and must be on the same Class C subnet as the IP address of the STAUT.

When the security parameter is set to WPA2-PSK, the test will use AES for unicast data and TKIP for multicast data. Running the test with security set to WPA-PSK will run the test using the TKIP cipher for both unicast and multicast data traffic. This configures the Test Engine using the default parameters defined in the *wifi_config_te<x>.tcl* file.

To run the test with different parameters, one or more of the optional command parameters may be used. For example, here is the command used to run a WPA test on channel 165 with an SSID of *my_company*, a pre-shared key of *this_key*, and to assign the ADEPT-WFA RF IP address of 192.168.8.5:

```
tmm_config_mic_test Azimuth-DEP-1 11a WPA-PSK 192.168.8.5 -channel 165 -
ssid my_company -PSK this_key
```

Running the STAUT MIC test requires the ADEPT-WFA hardware to be in AP mode. If any other type of test was run previously, the ADEPT-WFA will change from Client mode to AP mode and reboot automatically. The *tmm_config_staut_mic_test* command will not return until the testMAC Engine has finished rebooting and the testMAC Engine is reconnected. Configuring the particular test parameters on the ADEPT-WFA AP might require an additional reboot for certain configuration parameters to become active. If required, the configuration command uses *tmm_reboot*, which automatically reconnects to the Test Engine after the reboot has completed. This is a blocking function so the shell will hold in this state for approximately 45 to 60 seconds:

```
TestSTA B> reboot
:OK:00@
TestAP B>
%
```

The STAUT configuration command sets the ADEPT-WFA in AP mode and sets all required parameters including: the AP's band and channel, the SSID, security mode, pre-shared key passphrase (ASCII), and the ADEPT-WFA AP's IP address. It also enables the ADEPT-WFA AP's radio and starts beaconing. This leaves the ADEPT-WFA ready to run a MIC Countermeasure test. Upon completion of the configuration command, the MIC test parameters are displayed. An example of the test configuration output is:

```
TestAP B>
04/11/06:11:50:08 INFO MIC STAUT Countermeasure Test Configuration:
04/11/06:11:50:08 INFO Band = 11b; Channel = 11; PSK = 12345678;
ssid = wifi
%
```

The next set of steps prepare the STAUT and test bed devices:

- ❑ Configure the STAUT.
- ❑ Associate the STAUT to the ADEPT-WFA.
- ❑ Start a ping session from the STAUT to ADEPT (destination for ping is the <attackerIP> address entered in the *tmm_config_staut_mic_test* command).

The ADEPT-WFA STAUT MIC Test is started by using this command:

```
tmm_start_staut_mic_test Azimuth-DEP-1 <security> <staut MAC address>
```

The <security> parameter must match the parameter for which the ADEPT-WFA hardware was configured, either WPA-PSK or WPA2-PSK, or the test will not function properly. The ADEPT-WFA AP needs the STAUT MAC address to be entered for the DUT.

The STAUT MIC Test is run by a state machine in the ADEPT-WFA firmware. The timing between corrupted MIC packets and verification of reception of MIC failure reports are all handled by the Test Engine. Verification of STAUT receiving pings or not has to be checked on the STAUT. Status of the test is obtained by calling the status command:

```
tmm_get_staut_mic_test_status Azimuth-DEP-1 <security>
```

Test progress and final test results (either “Test complete” or “Test failed”) can be determined by parsing the returned status. The normal (passing) status progression for a test is:

- Ready for first attack
- First attack sent
- Second attack sent
- Third attack sent
- Waiting for disassociation
- Waiting for end of quiet period
- Waiting for reassociation
- Test complete

Note: Depending on the timing of status requests, it is possible to miss a particular response. If the test completes, fails, or is aborted (stopped before completion), the last state/status is maintained for retrieval until the test is restarted.

The test has failed if any of the following status responses are returned:

- Test failed. First attack sent. Unexpected deauthentication/disassociation.
- Test failed. First attack sent. No Mic report received.
- Test failed. Second attack sent. Unexpected deauthentication/disassociation.
- Test failed. Second attack sent. No Mic report received.
- Test failed. Third attack sent. No Mic report received.
- Test failed. Waiting for end of quiet period. Association received too soon.
- Test failed. Waiting for reassociation. No association received.*

The ADEPT-WFA firmware does not automatically fail a STAUT for ping or data frames received from the STAUT during the quiet period. After the test reaches the state "Waiting for end of quiet period ", The following command must be issued to retrieve counters that indicate if the STAUT is continuing to transmit ping or other data frames to the ADEPT-WFA AP:

```
tmm_get_staut_mic_counters Azimuth-DEP-1
```

The command returns the keyed list:

```
{dataPkts <number data frames>} {pingPkts <number of ping frames>}
```

The dataPkts value contains the total number of non-ping data frames received by the ADEPT AP (destination MAC address) and sent by the STAUT (source MAC address). For example, EAPOL data frames would be counted in the dataPkts. The pingPkts value contains the number of ICMP data frames received by the ADEPT AP (destination MAC address) and sent by the STAUT (source MAC address). Note that these counter values return the total number of frames received after the third MIC Failure Report until the STAUT reassociates to the ADEPT AP.

The ADEPT-WFA Compliance Tester retrieves the MIC counters at 10 second intervals beginning at the start of the quiet period until the STAUT reassociates to the ADEPT AP. This gives details of when these packets are transmitted. At the very least, these counters must be checked at test completion to get the total number of data and ping frames received during the quiet period.

Ideally, an AP should not receive any data frames from a STA that is not associated to it; non-ping data frames that are not actually attempting to forward data to another station have been allowed, but the ping packets (attempting to forward data to an end station), in general, are not allowed. If the pingPkt value is not zero, the test should be considered a failure.

The following partial script shows how to retrieve the number of data frames and ping frames from the ADEPT testMAC Engine:

- set counters [tmm_get_staut_mic_counters Azimuth-DEP-1]
- set dataFrames [key]get counters dataPkts]
- set pingFrames [key]get counters pingPkts]

Whether the test completes, fails, or is interrupted, the test should be stopped by using this command:

```
tmm_stop_staut_mic_test Azimuth-DEP-1 <security>
```

This command stops any pending state in the firmware, such as *ready to send corrupted MIC packet on next transmitted packet* and *disables the ADEPT-WFA AP's radio*. This is beneficial since it will stop the radio from beaconing preventing undesired/unwanted associations to the ADEPT-WFA AP when not running STAUT MIC tests.

Running the APUT MIC Countermeasure Test

This test requires more programming than other tests. Unlike the STAUT MIC test, the timing and states of the APUT MIC Countermeasure test are controlled by the Tcl script. The ADEPT-WFA operates in Client mode for the APUT; a ping session with the APUT must be started on the ADEPT-WFA client.

The first step is to configure the ADEPT-WFA for the test. There is only one ADEPT Client security mode for the APUT, WPA, so it is not a parameter to any of the APUT MIC commands. Connecting the ADEPT-WFA client with WPA security forces it to use the TKIP cipher which allows it to create frames with a corrupted MIC field. Configure ADEPT-WFA using this command:

```
tmm_config_aput_mic_test Azimuth-DEP-1 <band>
```

Replace <band> with the desired band to run the test on (either 11a, 11b, or 11g). To run the test with different parameters, one or more of the optional command parameters may be used.

Note: Due to scanning and association to the APUT, the channel and ssid are not valid until the ADEPT-WFA client is associated to the APUT. The channel and ssid are determined by the channel and ssid that the APUT is operating on. To run a WPA test on channel 165, the APUT has to be configured to operate on channel 165. To run the test with an SSID of *my_ssid* and a pre-shared key of *this_key*, the command would be:

Note: `tmm_config_apat_mic_test Azimuth-DEP-1 11a --ssid my_ssid --PSK this_key`

Running the APUT MIC test requires the ADEPT-WFA hardware to be in client mode. If the previous test was a STAUT MIC test, then the ADEPT-WFA will change from AP mode to Client mode and reboot. This configuration command uses the `tmm_reboot` command which automatically reconnects to the Test Engine after the reboot has completed. This is a blocking function so the shell will hold in this state for approximately 45 to 60 seconds:

```
TestAP B> reboot
:OK:00@
TestSTA B>
```

The final configuration step is to configure the ADEPT-WFA Client's IP address. This is accomplished by calling the `sta_set_ip_config` command as follows:

```
sta_set_ip_config Azimuth-DEP-1 -ip <IP address> -mask 255.255.255.0
```

Replace <IP address> with the IP address to assign to the ADEPT-WFA RF Port.

This address must be on the same subnet as the ping destination PC's IP address and must not be on the same subnet as the testMAC Engine Bus-net assigned IP address (the IP address used in the `connect_device` command).

The next step is to configure the APUT and testbed devices and start the ping sessions from the testbed STAs through the APUT (destination is a PC connected to the Ethernet side of the APUT).

When the system is set up and the ADEPT-WFA is configured for the test, associate the ADEPT-WFA client to the APUT. This consists of three steps:

- ❑ Scanning.
- ❑ Connecting to the APUT.
- ❑ Associating.

The ADEPT-WFA Client is made to scan a wireless band by calling this command:

```
tmm_scan Azimuth-DEP-1 <band> -arrayName <array>
```

The <band> parameter is replaced with the wireless band that the test was configured for and is being performed on (11a, 11b, or 11g). The parameter value <array> is the name of the array that will hold the results of the completed scan. This is a Tcl named array containing required values to choose the APUT and connect to it. The following is a list of the required parameters to search the list for the APUT and extract the BSSID (for complete list of `-arrayName` returned values and indexing refer to the `tmm_scan` command in Appendix C):

- ❑ `array(count)` - specifies the number of APs discovered on <band> during the scan.

- ❑ `array(i, mac)` - specifies the BSSID of the AP indexed by *i*.
- ❑ `array(i, ssid)` - specifies the SSID of the AP indexed by *i*.
- ❑ `array(i, chan)` - specifies the channel of the AP indexed by *i*.

The array can be searched from one to count for the desired AP. If the count returned is 0, or if the APUT is not found by searching the array, verify that the APUT setup and ADEPT-WFA are configured properly, and issue the TMM command scan again.

The command `tmm_scan` is a blocking command; it will not return until the ADEPT-WFA client has completed scanned all of the channels in the desired band. Depending on the scanned band, this can take several seconds to complete.

Connect the ADEPT-WFA Client to the AP using:

```
tmm_connect_ap Azimuth-DEP-1 <bssid>
```

The parameter `<bssid>` can be obtained from the named array using, `$array(i, mac)`, with *i* equal to the index of the APUT. An error will occur if the Test Engine is unable to connect to the APUT.

Once the ADEPT-WFA client is connected to the APUT, associate the client by calling this command:

```
tmm_associate_group Azimuth-DEP-1
```

The next step is to verify that the ADEPT-WFA Client has associated to the APUT. Association may take several seconds depending on the security mode and the APUT. Issue the following command, saving the return value.

```
set assoc [tmm_get_group_connected_client_num Azimuth-DEP-1]
```

If the ADEPT-WFA Client has successfully completed association and security authorization (completed EAPOL Key exchange), then return value will be 1. If the return value is 0, association and/or security key exchange has not completed yet. Since the completion of key exchange may take several seconds, depending on the APUT, this query for connection should be placed in a timed loop for about 10 seconds. An example of a timed loop including a 1 second delay between successive reading of the connected state follows:

```
set timeout 10
while {timeout} {
  set assoc [tmm_get_group_connected_client_num Azimuth-DEP-1]
  if {assoc} {
    break
  }
  incr timeout -1
  after 1000
}
```

After the ADEPT-WFA client is associated to the APUT, start a ping session using the `traffic_send_ping` command. Details of this command are listed in “[ADEPT-WFA Tcl Commands](#)” (on page B-1). The destination IP address must be on the same subnet as the ADEPT-WFA Client and not on the same subnet as the ADEPT-WFA control IP address (the address used in the `connect_device` command). The APUT MIC test is designed to use continuous pings in background (non-blocking) and using the ping callback feature of the `traffic_send_ping` command. Continuous pings are set when the duration parameter is set to

continuous. The callback feature is invoked by entering the argument *-callback 1*. The callback feature requires an additional parameter (*-arrayName <array>*) that specifies the name of the array containing the ping statistics returned when the pingcallback routine is called. When the *traffic_send_ping* is invoked with the callback feature, it returns immediately with a pingcallback function. The syntax of the *traffic_send_ping* command is:

```
traffic_send_ping Azimuth-DEP-1 <dest IP address> <ping_size> <duration>
-callback 1 -arrayName <array>
```

The named array *<array>* contains two indexed values (*array(send)* and *array(receive)*) that indicate the number of transmitted/received pings.

An example usage of this command in a script follows:

```
set ping_cb [traffic_send_ping Azimuth-DEP-1 192.168.8.35 1024
continuous -callback 1 -arrayName pingStats]
```

This will start a continuous ping session on the ADEPT-WFA Client to IP address 192.168.8.35 with ping packet data size equal to 1024 bytes. The variable *ping_cb* will contain the ping callback function. Ping statistics can easily be obtained by just calling this function as follows:

```
$ping_cb
```

Every time the callback function is called, the array pingStats will be updated. The number of pings transmitted equals *\$pingStats(send)* and the number of pings received equals *\$pingStats(receive)*. The ping callback returns 0, the ping session is still active. If the ping callback returns 1, the ping session has been stopped. To stop the ping session, issue the ping callback with stop as follows:

```
$ping_cb stop
```

Now that the system is set up, the ADEPT-WFA client is associated to the APUT and a ping session has been started, the MIC APUT Test can begin. Issuing the command *tmm_send_aput_mic_attack* sends a single MIC attack to the APUT. There are two types of attacks, corrupted MIC packets (*badmic*) and MIC Failure Reports (*failurereport*). The corrupted MIC packets are generated by corrupting the MIC of the next ping request transmitted to the APUT. A MIC Failure Report is transmitted by constructing an EAPOL data packet of type Failure report. An attack is sent by calling:

```
tmm_send_aput_mic_attack Azimuth-DEP-1 <type>
```

where *type* is replaced with either *badmic* or *failurereport*.

The Tcl script must properly time the sending of the MIC attacks and verify ADEPT-WFA client ping reception, or lack thereof, depending on the state of the test. The following is the basic flow of an APUT MIC test:

1. Send the first badmic MIC attack.
2. Delay for 70 seconds while verifying ping reception.
3. Send the second badmic MIC attack.
4. Delay for 50 seconds while verifying ping reception.
5. Send the third badmic MIC attack.
6. Verify that the ping reception has stopped.

7. Attempt to associate the ADEPT-WFA client to the APUT (this should fail).
8. Wait for 60 seconds after the third MIC attack and reassociate ADEPT-WFA client to APUT.
9. Verify ADEPT-WFA Client receives pings sent to APUT.
10. Repeat steps 1 through 9 using failurereport MIC attacks.

At the end of the test, issue the stop command. This is especially important if the test failed due to ping reception stopped or unexpected disassociation of the ADEPT-WFA client since there may be an impending MIC attack that needs to be cleared. The command to stop an APUT MIC attack is:

```
tmm_stop_aput_mic_attack Azimuth-DEP-1
```


Appendix B: ADEPT-WFA Tcl Commands

The following tables provide details concerning ADEPT-WFA Tcl commands:

connect_device

Tcl Command Description	Connects the control PC Tcl session to an ADEPT Test Engine. This function returns a target value that must be used in all other ADPET Tcl commands as the target value. This function validates that the connected device is an ADEPT-WFA and sets the Test Engine time of day to the controlling PC's time of day if an NTP server is present. This function is an Entry point to all other ADEPT-WFA functions.
Syntax	connect_device <sdkTarget> <ip>
Mandatory Argument Description/Options	<p>sdkTarget — specifies the ADEPT as the target for the Azimuth SDK. The SDK target for the ADEPT-WFA (basic) is “Azimuth-DEP.”</p> <p>Range: Azimuth-DEP</p> <p>Default Value: None</p> <p>ip — specifies the IP address of the ADEPT Test Engine to which to connect.</p> <p>Range: Valid IP address.</p> <p>Default Value: None</p> <p>Syntax: <XXX.XXX.XXX.XXX></p>
Return Value	Returns the target value of the connected device. The target value is required as the target parameter for all other commands. Example: Azimuth-DEP-1
Example	connect_device Azimuth-DEP 10.1.1.39

reconnect_device

Tcl Command Description	Reconnects to a previously connected ADEPT-WFA target. This command retains the target obtained from the “connect_device” (on page B-1) function. To maintain an existing target (such as in cases in which the target is being cached elsewhere in the user code), Azimuth recommends that this command be used.
Syntax	reconnect_device <target>

reconnect_device

Mandatory Argument Description/Options	target — specifies the target value returned from the original connect_device call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the Expect spawn ID of the Telnet session. This return value is used for debugging purposes only. Example: sock1768
Example	reconnect_device Azimuth-DEP-1

disconnect_device

Tcl Command Description	Disconnects/terminates a Telnet control session with an ADEPT-WFA target.
Syntax	disconnect_device <target>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	There is no return value.
Example	disconnect_device Azimuth-DEP-1

tmm_reboot

Tcl Command Description	Reboots the ADEPT-WFA Test Engine and then reconnects to that device. This command should be used when you want to reboot an ADEPT-WFA Test Engine and then reconnect to it to continue processing. The re-connection will keep the same hostId as the original connection (see “connect_device” (on page B-1) and “reconnect_device” (on page B-1)). This command blocks until the testMAC engine completes the reboot and is reconnected.
Syntax	tmm_reboot <target>
Mandatory Argument Description/Options	target - value returned from the original connect_device call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	There is no return value.
Example	tmm_reboot Azimuth-DEP-1

adept_reboot_only

Tcl Command Description	Reboots the ADEPT-WFA Test Engine. This routine does not automatically reconnect to the device. This is a non-blocking command, returning immediately after the reboot command is issued to the testMAC Engine.
Syntax	adept_reboot <target>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	There is no return value.
Example	adept_reboot Azimuth-DEP-1

tmm_set_console_ip

Tcl Command Description	Sets the host (controlling PC) IP address in the ADEPT-WFA Test Engine. This IP Address is used for the FTP server and as the destination for the System, Output, and FTP logs. The IP address must be the IP address of the controlling PC. The IP address must be on the same subnet as the ADEPT-WFA Test Engine’s Bus-Net IP address (the IP address used in the “connect_device” (on page B-1) command).
Syntax	tmm_set_console_ip <target> <ip>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X> ip — IP address of the host PC controlling the ADEPT-WFA Test Engine. Note: This IP address must be on the same subnet as the ADEPT-WFA Test Engine. Default Value: None Range: Valid IP address. Syntax: <XXX.XXX.XXX.XXX>
Return Value	There is no return value.
Example	tmm_set_console_ip Azimuth-DEP-1 192.168.2.1

ap_get_radio_client_list

Tcl Command Description	Obtains a list of the stations currently associated to the ADEPT-WFA MIC test AP (attacker). This is useful to verify that a STAUT is associated to the ADEPT-WFA AP and to obtain the STAUT's MAC address prior to starting a STAUT MIC Attack Test. The STAUT MAC address is a required parameter for the <code>"tmm_start_staut_mic_test"</code> (on page B-16) procedure.
Syntax	<code>ap_get_radio_client_list <target> [radioId]</code>
Mandatory Argument Description/Options	target — value returned from the original <code>"connect_device"</code> (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Optional Argument Description/Options	radioId — dummy radioId, used to indicate a specific radio in multi-radio APs. <i>This function is not applicable to the ADEPT-WFA and should not be used for the ADEPT-WFA; i.e., leave the default value of this argument as null.</i> Default Value: ""(i.e., null) Syntax: None
Return Value	Returns a list of stations associated to the ADEPT testMAC Engine AP. If there are no associated stations, this returns null. Example: 00:06:2A:FB:AB:23, 00:06:50:12:FC:0B
Example	<code>ap_get_radio_client_list Azimuth-DEP-1</code>

tmm_scan

Tcl Command Description	Obtains a list of APs (BSSIDs) that the ADEPT-WFA MIC Test station discovers through scanning. This command is used to support the APUT MIC countermeasures tests. This function blocks until the ADEPT-WFA's client completes scanning. Note: The ADEPT-WFA performs active scanning only.
Syntax	<code>tmm_scan <target> <band> -arrayName <name></code>

tmm_scan (Continued)

Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call.</p> <p>Default Value: None</p> <p>Syntax: Azimuth-DEP-<X></p> <hr/> <p>band — specifies whether to scan a single wireless mode or all wireless modes. The STAUT MIC tests are defined to run in a specific band. The desired band to test should be input as the band parameter.</p> <p>Default Value: None</p> <p>Syntax: 11a, 11b, 11g, all</p> <hr/> <p>-arrayName <name> — specifies a named array containing the BSS info for APs discovered during a scan. The array (count) value indicates the number of APs discovered during the scan. All other BSS information is indexed by the name of the desired value AND the index/number that the BSS was discovered.</p> <p>Example:</p> <pre><name>(1,band) = 11b <name>(1,bssType) = INFRASTRUCTURE <name>(1,chan) = 11 <name>(1,mac) = 00:05:9A:38:73:F5 <name>(1,rsi) = 56 <name>(1,ssid) = wifi <name>(1,status) = NOT_CONNECTED <name>(1,supportedRates) = 1 2 5.5 11 <name>(1,timOffset) = 0 <name>(count) = 1</pre>
Return Value	Returns 0 upon successful completion and nonzero upon error. List of BSSIDs discovered are returned through <name> array argument (see above).
Example	tmm_scan Azimuth-DEP-1 -arrayName bssList

tmm_connect_ap

Tcl Command Description	Joins the ADEPT-WFA to an APUT for the APUT MIC countermeasures test. It is required to call this command before attempting to associate the ADEPT-WFA client to the APUT with the “ tmm_associate_group ” (on page B-6) function.
Syntax	tmm_connect_ap <target> <apBssid>

tmm_connect_ap (Continued)

Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call . Default Value: None Syntax: Azimuth-DEP-<X>
	apBssid — specifies the MAC address of the DUT. Default Value: None Syntax: <AA:BB:CC:DD:EE:FF>
Return Value	There is no return value.
Example	tmm_connect_ap Azimuth-DEP-1 00:00:CA:4B:90:A2

tmm_associate_group

Tcl Command Description	Associates the ADEPT-WFA to an APUT for the APUT MIC countermeasures test. The “tmm_connect_ap” (on page B-5) function must be called prior to calling this function. This function will return an error if the ADEPT-WFA client is not able to associate to the APUT.
Syntax	tmm_associate_group <target> [ssid]
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Optional Argument Description/Options	ssid — specifies the SSID of the APUT to which to associate. Since the “tmm_connect_ap” (on page B-5) function specifies the AP with which the ADEPT-WFA station will associate. <i>This argument is not used for the ADEPT-WFA.</i> Default Value: “” Syntax: string
Return Value	There is no return value.
Example	tmm_associate_group Azimuth-DEP-1

tmm_get_group_connected_client_num

Tcl Command Description	Associates the ADEPT-WFA to an AP for the APUT MIC countermeasures test. The "tmm_connect_ap" (on page C-6) function must be called prior to calling this function. This function will return an error if the ADEPT-WFA client is not able to associate to the APUT.
Syntax	tmm_associate_group <target>

tmm_get_group_connected_client_num (Continued)

Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns 1 if the ADEPT-WFA testMAC Client is associated and has successfully completed security protocol key exchange. Return value of 0 indicates that either association and security has failed or has not completed.
Example	tmm_get_group_connected_client_num Azimuth-DEP-1

sta_set_ip_config

Tcl Command Description	Sets the IP configuration of the specified station for the APUT MIC countermeasures test. This IP address becomes the source address for the ping session from the ADEPT-WFA station to the APUT.
Syntax	sta_set_ip_config <target> <-ip <IP Address>> <-mask <IP Mask>>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X> -ip <IP Address> — specifies the IP address of the ADEPT-WFA Test Engine to which to connect. Default Value: None Range: Recommended IP Mask - 255.255.255.0 Syntax: -ip <XXX.XXX.XXX.XXX> -mask <IP Mask>— specifies the static IP address mask to use for the ADEPT-WFA station. Default Value: None Range: valid static IP mask Syntax: -mask <XXX.XXX.XXX.XXX>
Return Value	There are no return values.
Example	sta_set_ip_config Azimuth-DEP-1 -ip 192.168.1.68 -mask 255.255.255.0

traffic_send_ping

<p>Tcl Command Description</p>	<p>Starts traffic by sending pings from the ADEPT-WFA station (source device) to the APUT (destination device) for the ADEPT-WFA APUT MIC countermeasures test. The ADEPT-WFA MIC countermeasures tests are designed to use the callback feature of this command with continuous pings.</p> <p>Note: The ADEPT Client IP address must be configured using the <code>sta_set_ip_config</code> command prior to transmitting pings.</p>
<p>Syntax</p>	<p><code>traffic_send_ping <target> <dstIP> <frameSize> <duration> [-callback <value>] [-arrayName pingData]</code></p>
<p>Mandatory Argument Description/Options</p>	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>dstIP — specifies the ping destination IP address of the APUT. Default Value: None Syntax: <XXX.XXX.XXX.XXX> Example: <code>traffic_send_ping dstIp 192.168.3.14 ..</code></p> <p>frameSize — specifies the frame size of the ping. The ADEPT-WFA uses a frameSize equal to 1024. Default Value: None. Syntax: <value> Example: 1024</p> <p>duration - specifies the number of seconds to ping. The testMAC Engine ping rate is approximately 1 ping per second.</p> <p>Note: The ADEPT-WFA APUT MIC Test is designed to use continuous pings. Default Value: None Syntax: <value> Example: continuous</p>

traffic_send_ping (Continued)

Optional Argument Description/ Options	<p>-callback <0 or 1> — specifies whether traffic_send_ping is a blocking function until ping duration has completed or whether it returns immediately returning the callback function to retrieve ping statistics. When callback is set to 0, the function returns the number of received pings after the duration of pings has completed (blocking). When the callback is set to 1, the function returns a dynamic macro immediately (non-blocking). The callback routine is used to retrieve the number of pings sent and received at the time of query. The ADEPT-WFA is designed to use the non-blocking callback.</p> <p>Default Value: 0</p> <p>Syntax: 0 or 1</p> <p>Example: traffic_send_ping Azimuth-DEP-1 192.168.1.135 1024 continuous -callback 1 -arrayName pingData</p> <p>-arrayName — specifies the array name to store the ping statistics (send or receive) or when the callback option is used.</p> <p>Default Array: None</p> <p>Example: traffic_send_ping Azimuth-DEP-1 192.168.4.10 1024 30 -callback 1 -arrayName pingData</p>
Return Value	<p>If callback = 1, returns a dynamic macro called <i>pingcallback_<timestamp></i></p> <p>If callback = 0, returns number of received pings</p> <p>If pingcallback = 0, process ping is still running</p> <p>If pingcallback = 1, ping is completed</p> <p>If pingcallbackstop, stop ping session, callback returns 1</p>
Example	<pre>set ping_cb [traffic_send_ping Azimuth-DEP-1 192.168.1.65 1024 continuous -callback 1 -arrayName pingData] \$ping_cb parray pingData pingData(send) = 525 pingData(receive) = 519 \$ping_cb stop - Stop ping session</pre>

traffic_stop_ping

Tcl Command Description	Stops current running ping session. This is an alternative command to stop pings (to pingcallback described under “traffic_send_ping” (on page B-8)).
Syntax	traffic_stop_ping <target>

traffic_stop_ping (Continued)

Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	There is no return value.
Example	traffic_stop_ping Azimuth-DEP-1

tmm_config_nav_test

Tcl Command Description	Configures the ADEPT-WFA for a WFA NAV Test. This procedure will configure the ADEPT-WFA’s wireless band and channel, set security to open, and disable fragmentation and RTS thresholds. It will also configure the ADEPT-WFA to use RF Port B and set the attenuator to minimum. Additionally, it will check that the ADEPT-WFA is in client mode. If not, it will set the ADEPT-WFA testMAC Engine to client mode and reset the module. An error occurs if any configuration command is not completed successfully.
Syntax	tmm_config_nav_test <target> <band> [-channel<channel number>]
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>band — specifies the wireless mode of the radio. Default Value: None Syntax: <11a 11b 11g></p> <p>-channel <channel number> — specifies the channel on which to run the NAV test. Default Value: channel specified by the WFA test specification based on the selected band (11a=36; 11b,11g=11) as read from the test configuration file <i>wifi_config_<te>.tcl</i>. Syntax: -channel <XX> (ex. -channel 52)</p>
Return Value	There is no return value.
Example	tmm_config_nav_test Azimuth-DEP-1 11g -channel 11

tmm_start_nav_test

Tcl Command Description	Starts transmission of CTS NAV test packets on a preconfigured radio band/channel. The function “ tmm_config_nav_test ” (on page B-10) must be called prior to starting a NAV Test. The ADEPT-WFA NAV test sends CTS to self packets with a duration of XXX at a rate of YYY.
Syntax	tmm_start_nav_test <target>
Mandatory Argument Description/Options	target - value returned from the original “ connect_device ” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the status of an ADEPT-WFA NAV start: “WFA Nav Test Started” “WFA NAV Test Already Running”
Example	tmm_start_nav_test Azimuth-DEP-1

tmm_stop_nav_test

Tcl Command Description	Stops transmission of CTS NAV packets.
Syntax	tmm_stop_nav_test <target>
Mandatory Argument Description/Options	target — target of the target device. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the status of an ADEPT-WFA NAV Stop: “WFA NAV Test Halted” "WFA NAV Test Not Running"
Example	tmm_stop_nav_test Azimuth-DEP-1

tmm_get_nav_test_status

Tcl Command Description	Obtains the current status of a CTS NAV test.
Syntax	tmm_get_nav_test_status <target>
Mandatory Argument Description/Options	target - value returned from the original “ connect_device ” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>

tmm_get_nav_test_status (Continued)

Return Value	Returns the status of an Adept NAV test: "Not Running" "Running"
Example	tmm_get_nav_test_status Azimuth-DEP-1

tmm_config_plcp_test

Tcl Command Description	Configures the ADEPT-WFA for a WFA PLCP Test. This procedure will configure the ADEPT-WFA's wireless band and channel, set security to open, and disable fragmentation and RTS thresholds. It also configures the ADEPT-WFA to use RF Port B and set the attenuator to minimum. Additionally, it will check that the ADEPT-WFA is in client mode. If not, it will set the ADEPT-WFA Test Engine to client mode and reset the module. An error will occur if any configuration command is not completed successfully.
Syntax	tmm_config_plcp_test <target> <band> [-channel <channel number>]
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>band — specifies the wireless mode of the radio. Default Value: None Syntax: <11a 11b 11g></p> <p>-channel <channel number> — specifies the channel on which to run the NAV test. Default Value: channel specified by the WFA test specification based on the selected band (11a=36; 11b,11g=11) as read from the test. configuration file <i>wifi_config_<te>.tcl</i>. Syntax: -channel <XX> (ex. -channel 1)</p>
Return Value	There is no return value.
Example	tmm_config_plcp_test Azimuth-DEP-1 11b -channel 11

tmm_start_plcp_test

Tcl Command Description	Starts transmission of PLCP test packets on preconfigured radio band/channel. “tmm_config_plcp_test” (on page B-12) must be called prior to starting a PLCP Test. The ADEPT-WFA PLCP test transmits packets of size XX but stops packet transmission (by attenuating output by 60 dB) after a couple of milliseconds.
Syntax	tmm_start_plcp_test <target>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the status of an ADEPT-WFA PLCP Start: “WFA PLCP Test Started” “WFA PLCP Test Already Running”
Example	tmm_start_plcp_test Azimuth-DEP-1

tmm_stop_plcp_test

Tcl Command Description	Stops the transmission of PLCP test packets.
Syntax	tmm_stop_plcp_test <target>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the status of an ADEPT PLCP Stop: “WFA PLCP Test Halted” "WFA PLCP Test Not Running"
Example	tmm_stop_plcp_test Azimuth-DEP-1

tmm_get_plcp_test_status

Tcl Command Description	Obtains the current status of a PLCP test.
Syntax	tmm_get_plcp_test_status <target>

tmm_get_plcp_test_status (Continued)

Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call.</p> <p>Default Value: None</p> <p>Syntax: Azimuth-DEP-<X></p>
Return Value	<p>Returns the status of an Adept PLCP Test:</p> <p>"Not Running"</p> <p>"Running"</p>
Example	tmm_get_plcp_test_status Azimuth-DEP-1

tmm_config_staut_mic_test

Tcl Command Description	<p>Configures the ADEPT-WFA for a STAUT WFA MIC countermeasures test. This procedure sets up the ADEPT-WFA band, channel, security mode, PSK, and SSID for the test. If the ADEPT-WFA is in client mode, it will be changed to AP mode and rebooted. Additionally the ADEPT-WFA may need to be rebooted for the configured parameters to take effect. An error occurs if any configuration command is not completed successfully.</p>
Syntax	<p>tmm_config_staut_mic_test <target> <band> <security> <DUT> <attackerIp> [-channel <channel number>] [-PSK <Pre-Shared Key>] [-ssid <“string”>]</p>

tmm_config_staut_mic_test (Continued)

<p>Mandatory Argument Description/Options</p>	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X> Example: Azimuth-DEP-1</p> <hr/> <p>band — specifies the wireless mode of the radio. Default Value: None Syntax: <11a 11b 11g></p> <hr/> <p>security — specifies the security mode for the MIC countermeasures test. Default Value: None Syntax: <WPA WPA2></p> <hr/> <p>attackerIp — specifies the ADEPT AP IP address for the STAUT. This must not be on the same subnet as the Test Engine’s Bus-Net IP address. This IP address must be used for the ping destination for the ping session started on the STAUT. Default Value: None Syntax: <xxx.xxx.xxx.xxx> Example: 192.168.1.135</p>
<p>Mandatory Argument Description/Options (<i>continued</i>)</p>	<p>-channel <channel number> — specifies the channel on which to run the MIC countermeasures test. Default Value: channel specified by the WFA test specification based on the selected band (11a=36; 11b, 11g=11) as read from the test configuration file <i>wifi_config<x>.tcl</i> (where x= 1 or 2). Syntax: -channel <channel number> (ex. -channel 52)</p> <hr/> <p>-PSK <Pre-Shared Key> — specifies the Pre-Shared Key for the authentication for the MIC test. Default Value: PSK specified by WFA test as read from the test configuration file <i>wifi_config<x>.tcl</i> (where x=1 or 2). Syntax: -PSK <"string"></p> <hr/> <p>-ssid <"string"> — specifies the SSID for the MIC countermeasures test. Default Value: SSID specified by WFA test as read from the test configuration file <i>wifi_config<x>.tcl</i> (where x= 1 or 2). Syntax: <"string"></p>

tmm_config_staut_mic_test (Continued)

Return Value	There is no return value.
Example	tmm_config_staut_mic_test Azimuth-DEP-1 11a WPA2-PSK APUT 192.168.1.135

tmm_start_staut_mic_test

Tcl Command Description	<p>Starts the MIC attack test. Prior to starting a MIC attack test, “tmm_config_staut_mic_test” (on page B-14) must be called to properly configure the ADEPT-WFA for the STAUT MIC test. This function must be called with the same security parameter as was used in the configuration function. This procedure returns immediately after initializing test status and starting the ADEPT STAUT MIC state machine. The ADEPT-WFA firmware times and controls this test. The status of the test is obtained with the function “tmm_get_staut_mic_test_status” (on page B-17).</p> <p>Note: Only one MIC attack test (either with WPA or WPA2) can be run at any given time. An error is generated if any ADEPT control command called by this routine is not completed successfully. Possible errors include “Station Not Found” and “WFA MIC Test Already Running.”</p>
Syntax	tmm_start_staut_mic_test <target> <security> <mac>
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>security — specifies the security mode for the MIC countermeasures test. Default Value: None Syntax: <WPA-PSK WPA2-PSK></p> <p>mac — specifies the MAC address (BSSID) of the APUT. Default Value: None Syntax: <AA:BB:CC:DD:EE:FF></p>
Return Value	There is no return value.
Example	tmm_start_staut_mic_test Azimuth-DEP-1 WPA-PSK 00:00:CA:4B:90:A2

tmm_stop_staut_mic_test

Tcl Command Description	Stops a STAUT MIC attack test. This command should be called to stop or interrupt a STAUT MIC test. This command will clear any pending attack and reset the ADEPT-WFA STAUT state machine. This routine does not clear the last test status, leaving it available for retrieval after the test has been stopped.
Syntax	tmm_stop_staut_mic_test <target> <security >
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>security — specifies the security mode for the MIC countermeasures test. Default Value: None Syntax: <WPA-PSK WPA2-PSK></p>
Return Value	There is no return value.
Example	tmm_stop_staut_mic_test Azimuth-DEP-1 WPA2-PSK

tmm_get_staut_mic_test_status

Tcl Command Description	Obtains the current state and status of a STAUT MIC countermeasures test.
Syntax	tmm_get_mic_test_status <target> <security>
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X></p> <p>security — Specify the security protocol for the STAUT MIC countermeasures test. Default Value: None Syntax: <WPA-PSK WPA2-PSK></p>

tmm_get_staut_mic_test_status (Continued)

Return Value	<p>Returns the current state of the STAUT MIC countermeasures test. Possible values include:</p> <ul style="list-style-type: none"> • “Ready for first attack.” • “First attack sent.” • “Second attack sent.” • “Third attack sent.” • “Waiting for disassociation.” • “Waiting for end of quiet period.” • “Waiting for reassociation.” • “Test completed.” • “Test failed. First attack sent. Unexpected deauthentication/disassociation.” • “Test Failed. Waiting for disassociation. No disassociation received.” • “Test failed. Waiting for the end of quiet period. Association received too soon.” • “Test failed. Waiting for reassociation. No association received.”
Example	tmm_get_staut_mic_test_status Azimuth-DEP-1 WPA-PSK

tmm_get_staut_mic_counters

Tcl Command Description	<p>Obtains the number of non-ping data packets and ping packets from the STAUT to the ADEPT-WFA AP. These counters begin counting data packets received from the STAUT after the third MIC failure report is received from the STAUT (the beginning of the quiet period) and continues counting until the STAUT reassociates to the ADEPT-WFA AP. During the quiet period, this command will return the received packet counts at the time of issue. These counters remain valid after the test has either Completed or Failed, allowing them to be retrieved. The STAUT MIC counters are reset when the STAUT MIC test is started.</p>
Syntax	tmm_get_staut_mic_counters <target>
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call.</p> <p>Default Value: None</p> <p>Syntax: Azimuth-DEP-<X></p>
Return Value	<p>Returns a keyed list containing the number of non-ping data packets and ping packets.</p> <p>Syntax: {dataPkts <value>} {pingPkts <value>}</p>
Example	tmm_get_staut_mic_counters Azimuth-DEP-1

tmm_config_aput_mic_test

Tcl Command Description	Configures the ADEPT-WFA for a WFA APUT MIC countermeasures test. This procedure sets up the ADEPT-WFA band, PSK, and SSID for the test. If the ADEPT-WFA is in AP mode, it will be changed to client mode and rebooted. An error occurs if any configuration command is not completed successfully.
Syntax	tmm_config_aput_mic_test <target> <band> [-PSK <"string">] [-ssid <"string">]
Mandatory Argument Description/Options	<p>target - value returned from the original "connect_device" (on page B-1) call.</p> <p>Default Value: None</p> <p>Syntax: Azimuth-DEP-<X></p>
	<p>band — specifies the wireless mode of the radio.</p> <p>Default Value: None</p> <p>Syntax: <11a 11b 11g></p>
Optional Argument Description/Options	<p>-PSK <Pre-Shared Key> — specifies the Pre-Shared Key for the authentication for the MIC test.</p> <p>Default Value: PSK specified by the WFA test as read from the test configuration file <i>wifi_config<x>.tcl</i> (where x= 1 or 2).</p> <p>Syntax: -PSK <"string"></p>
	<p>-ssid <"string"> — specifies the SSID for the MIC countermeasures test.</p> <p>Default Value: SSID specified by WFA test as read from the test configuration file <i>wifi_config<x>.tcl</i> (where x= 1 or 2).</p> <p>Syntax: -ssid <"string"></p>
Return Value	There is no return value.
Example	tmm_config_aput_mic_test Azimuth-DEP-1 11a

tmm_send_aput_mic_attack

Tcl Command Description	<p>Sends one attack packet, either a corrupted MIC as the response to the next ping request packet received, or a MIC Failure Report (EAPOL data packet). To properly configure the ADEPT-WFA for the APUT MIC test, "tmm_config_aput_mic_test" (on page C-16) must be called prior to sending an attack. This routine is non-blocking and returns immediately after issued.</p> <p>Note: Depending on the <type> argument, this command sends an individual MIC attack that is either a corrupted MIC in a ping packet or a MIC failure report.</p>
Syntax	tmm_send_aput_mic_attack <target> <type>
Mandatory Argument Description/Options	<p>target - value returned from the original “connect_device” (on page B-1) call.</p> <p>Default Value: None</p> <p>Syntax: Azimuth-DEP-<X></p> <p>type - specifies the type of attack to send to the APUT. It can either be a corrupted MIC in a ping packet (badmic) or a MIC failure report (failurereport).</p> <p>Default Value: None</p> <p>Syntax: <badmic failurereport></p>
Return Value	There is no return value.
Example	tmm_send_aput_mic_attack Azimuth-DEP-1 badmic

tmm_stop_aput_mic_attack

Tcl Command Description	<p>Stops an APUT MIC attack. This command should be called when interrupting or stopping an APUT MIC test. This command will clear any pending corrupted MIC in a ping packet (badmic) or MIC failure report (failurereport). This may be required if the test was halted due to ping session interruption (the ADEPT-WFA may still be waiting to receive a ping packet to corrupt the MIC).</p>
Syntax	tmm_stop_aput_mic_attack <target>

tmm_stop_aput_mic_attack

Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	There is no return value.
Example	tmm_stop_aput_mic_attack Azimuth-DEP-1

tmm_get_aput_mic_test_status

Tcl Command Description	Obtains the status of a MIC countermeasures test. This command returns the number of corrupted MIC packets and the number of Failure Reports sent.
Syntax	tmm_get_aput_mic_test_status <target>
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns the current status of an APUT MIC attack. Return values include the following: Corrupted MIC packets sent = x Failure Reports sent = y x,y = 0...3 Proper test sequence results in the following status return values: c = 0; F = 0 c = 1; F = 0 c = 2; F = 0 c = 3; F = 0 c = 3; F = 1 c = 3; F = 2 c = 3; F = 3
Example	tmm_get_aput_mic_test_status Azimuth-DEP-1

adept_get_aput_mic_config

Tcl Command Description	Obtains the test configuration for an APUT MIC Countermeasure test. This command retrieves and displays the band, channel, PSK, and SSID of the test. To retrieve the correct channel number for the test, this command must be called after the ADEPT-WFA client has been associated to the APUT.
Syntax	adept_get_aput_mic_config Azimuth-DEP-1
Mandatory Argument Description/Options	target - value returned from the original “connect_device” (on page B-1) call. Default Value: None Syntax: Azimuth-DEP-<X>
Return Value	Returns a list containing the following information: band, channel, PSK, and ssid Example: 11b 11 12345678 wifi
Example	adept_get_aput_mic_config Azimuth-DEP-1

Index

A

adept_reboot_only, [B-3](#)
ADEPT-WFA Test Engine, Properties of, [1-9](#)
ADEPT-WFA tests, configuring and running, [1-11](#)
ap_get_assoc, [B-4](#)
APUT MIC Countermeasure Test, running using Tcl scripts, [A-7](#)
Assigning a Management IP Address, [1-1](#)

C

Configuring and running the MIC test, [1-17](#)
Configuring and running the NAV test, [1-11](#)
Configuring and running the PLCP test, [1-14](#)
connect_device, [B-1](#)

D

DHCP Settings, modifying, [1-4](#)
disconnect_device, [B-2](#)

L

Logs
 configuring, [1-10](#)
 viewing, [1-10](#)

M

Managing ADEPT-WFAs, [1-2](#)
Managing Test Engines, [1-2](#)
MIC test, configuring and running, [1-17](#)

N

NAV test, configuring and running, [1-11](#)
NAV Test, running using Tcl scripts, [A-3](#)

P

PLCP test, configuring and running, [1-14](#)
Properties
 ADEPT-WFA Test Engine, [1-9](#)

R

reconnect_device, [B-1](#)
Restarting the ADEPT-WFA Test Engine, [1-7](#)

S

sta_set_ip_config, [B-7](#)
STAUT MIC Countermeasure Test, running using Tcl scripts, [A-4](#)

T

Tcl
 initiating a session, [A-2](#)
 Using Tcl scripts to run a NAV Test, [A-3](#)
 Using Tcl scripts to run a PLCP Test, [A-3](#)
 Using Tcl scripts to run a STAUT MIC Countermeasure Test, [A-4](#)
 Using Tcl scripts to run an APUT MIC Countermeasure Test, [A-7](#)

tmm__scan, [B-4](#)
tmm_associate_group, [B-6](#)
tmm_config_aput_mic_test, [B-19](#)
tmm_config_nav_test, [B-5](#), [B-10](#)
tmm_config_plcp_test, [B-12](#)
tmm_config_staut_mic_test, [B-14](#)
tmm_get_nav_test_status, [B-11](#)
tmm_get_plcp_test_status, [B-13](#)
tmm_get_staut_mic_test_status, [B-17](#)
tmm_reboot, [B-2](#)
tmm_set_console_ip, [B-3](#)
tmm_start_nav_test, [B-11](#)
tmm_start_plcp_test, [B-13](#)
tmm_start_staut_mic_test, [B-16](#)
tmm_stop_nav_test, [B-11](#)
tmm_stop_plcp_test, [B-13](#)
tmm_stop_staut_mic_test, [B-17](#)
traffic_send_ping, [B-8](#)
traffic_stop_ping, [B-9](#)

U

Updating the ADEPT-WFA License Key, [1-6](#)
Upgrading the ADEPT-WFA, [1-5](#)

