



ZEBRA

User Manual

Dart UWB Vision Reader

© 2013 Zebra Technologies Corp.

The copyrights in this manual and the software and/or firmware and hardware described therein are owned by Zebra Technologies (Zebra). Unauthorized reproduction of this manual or the software and/or firmware may result in imprisonment of up to one year and fines of up to \$10,000 (17 U.S.C.506). Copyright violators may be subject to civil liability.

ZEBRA and the Zebra logo are registered trademarks of ZIH Corp.

All other brand names, product names, or trademarks belong to their respective holders.

Proprietary Statement/Use

This document contains proprietary information of Zebra which may not be used, reproduced, or disclosed to any other parties for any other purpose without the express, written permission of Zebra. This document has been made available as part of the license that has been granted to an authorized user of Zebra software. It is intended solely for the information and use of parties operating and maintaining the equipment described herein. Use of this documentation is subject to the terms and limitations of that license agreement. This document describes all functionality that can be licensed for this product. Not all functionality described in this document may be available to you depending on your license agreement. If you are not aware of the relevant terms of your license agreement, contact sales at Zebra.

Product Improvements

Continuous improvement of products is a policy of Zebra Technologies. All specifications and designs are subject to change without notice.

Liability Disclaimer

Zebra takes steps to ensure that its published documentation is correct; however, errors do occur. Zebra reserves the right to correct any such errors and disclaims liability resulting there from.

Limitation of Liability

In no event shall Zebra, any of its licensors or anyone else involved in the creation, production, or delivery of the accompanying product (including hardware and software) be liable for any of the following (collectively referred to as "Injuries"): injuries (including death) or damages to persons or to property, or damages of any other kind, direct, indirect, special, exemplary, incidental or consequential, including, but not limited to, loss of use, lost profits, lost revenues, loss of data, business interruption, replacement costs, debt service or rental payments, or damages owing by you to others, whether arising out of contract, tort, strict liability or otherwise, arising from or relating to the design, use (or inability to use) or operation of these materials, the software, documentation, hardware, or from any services provided by Zebra (whether or not Zebra or its licensors knew or should have known of the possibility of any such Injuries) even if a remedy set forth herein is found to have failed of its essential purpose. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Typographical Conventions



Warnings call attention to a procedure or practice that could result in personal injury if not correctly performed. Do not proceed until you fully understand and meet the required conditions.



Cautions call attention to an operation procedure or practice that could damage the product, or degrade performance if not correctly performed. Do not proceed until understanding and meeting these required conditions.

Note

Notes provide information that can be helpful in understanding the operation of the product.

Table of Contents	Page
1. DOCUMENT OVERVIEW	7
2. PRODUCT DESCRIPTION AND FEATURES	7
3. PRODUCT SPECIFICATIONS	9
4. INSTALLATION AND MOUNTING.....	10
4.1 ITEMS REQUIRED (SUPPLIED BY INSTALLER)	11
4.2 TOOLS REQUIRED /SUGGESTED	11
4.3 CONNECTIONS	12
4.4 NETWORK AND POWER OPTIONS	14
5. CONFIGURATION	15
5.1 CONFIGURATION CONNECTION	15
5.2 UNIT CONFIGURATION	18
5.3 LAN WI-FI CLIENT CONFIGURATION	24
6. SUPPLICANT CONFIGURATIONS.....	34
6.1 PERSONAL CONFIGURATIONS.....	34
6.2 EXTENSIBLE AUTHENTICATION PROTOCOL (EAP).....	37
6.3 DIGITAL CERTIFICATES	39
6.4 COMMON EAP METHODS	42
6.5 ACTIVATING IPV6 TRANSPORT	51
6.6 FIRMWARE UPGRADE PROCESS.....	54
6.7 READ RANGE.....	57
7. OUTPUT DATA DESCRIPTION	63
7.1 DART VISION READER OUTPUT (PORT 5117)	63
7.2 DART VISION READER READ RANGE PORT (PORT 5110).....	64
7.3 DART VISION READER Z-SLMP OUTPUT (PORT 5118)	65
7.4 DART VISION READER SLMP OUTPUT (PORT 5119).....	66
8. REGULATORY COMPLIANCE INFORMATION	67
9. REGULATORY COMPLIANCE INFORMATION WLM54AG MODULE.....	68
APPENDIX A	70
A.1 MOUNTING INSTRUCTIONS.....	70
APPENDIX B	74
B.1 CONNECTOR REMOVAL.....	74
B.2 CABLE PREPARATION.....	75
APPENDIX C	80
REFERENCE: ISO 24730-1 FIELD DEFINITIONS AND DESCRIPTIONS.....	80
REFERENCE: ZEBRA FIELD DEFINITIONS AND DESCRIPTIONS USED BY Z-SLMP	81
APPENDIX D: DART VISION READER COUNTRY CODE KEY SETTING	83
APPENDIX E: WPA_SUPPLICANT SETTINGS DETAILS.....	88

Table of Figures	Page
FIGURE 1: HIGH-GAIN DART VISION READER	7
FIGURE 2: DVR CONNECTIONS (<i>NOTE: RJ-45 BACK SHELLS REMOVED IN FIGURE</i>)	13
FIGURE 3: NETWORK AND POWER	14
FIGURE 4: DAISY CHAIN CONNECTION	14
FIGURE 5: SSH FILES	17
FIGURE 6: SSH MAIN MENU	18
FIGURE 7: MAIN MENU	24
FIGURE 8: WLAN MENU	25
FIGURE 9: WLAN ON	25
FIGURE 10: MAIN MENU COUNTRY CODE	26
FIGURE 11: MANUFACTURING MENU	27
FIGURE 12: COUNTRY CODE LIST	27
FIGURE 13: SET COUNTRY CODE.....	28
FIGURE 14: SAVE SETTINGS	29
FIGURE 15: ANTENNA TYPE.....	29
FIGURE 16: FILE UPLOAD	30
FIGURE 17: ADVANCE MENU	31
FIGURE 18: GET REMOTE FILE.....	32
FIGURE 19: 802.11 CLIENT MENU	32
FIGURE 20: WPA_SUPPLICANT AT BOOT	33
FIGURE 21: WPA CONFIRMATION.....	33
FIGURE 22: EAP AUTHENTICATION.....	38
FIGURE 23: CERTIFICATE UPLOAD	39
FIGURE 24: CERTIFICATE DATA.....	40
FIGURE 25: SET VALID DATE.....	41
FIGURE 26: IPV6 EXAMPLE.....	52
FIGURE 27: FIRMWARE UPGRADE FILES.....	53
FIGURE 28: FIRMWARE UPGRADE MAIN MENU.....	54
FIGURE 29: FIRMWARE UPGRADE MENU.....	54
FIGURE 30: FIRMWARE UPGRADE SCRIPT.....	55
FIGURE 31: FIRMWARE UPGRADE RESPONSE.....	55
FIGURE 32: FIRMWARE UPGRADE COMPLETE	56
FIGURE 33: FIRMWARE UPGRADE SERIAL.....	56
FIGURE 34: READ RANGE MAIN MENU	57
FIGURE 35: READ RANGE MENU	58
FIGURE 36: READ RANGE DIRECT ENTRY.....	58
FIGURE 37: READ RANGE VALUE	59
FIGURE 38: READ RANGE INCREMENT/DECREMENT	59
FIGURE 39: READ RANGE SAVE	60
FIGURE 40: FINE READ RANGE	60
FIGURE 41: READ RANGE FINE SET	61
FIGURE 42: READ RANGE FINE SAVE	61
FIGURE 43: READ RANGE COURSE DISTANCE.....	62
FIGURE 44: READ RANGE FINE DISTANCE.....	62
FIGURE 45: PORT 5110 COMMANDS.....	64
FIGURE 46: INSTALLATION TO HORIZONTAL METAL STRUT.....	70
FIGURE 47: VERTICAL MOUNTING TO METAL STRUT.....	70
FIGURE 48: OPTIONAL MOUNTING BRACKET (UM-120-00)	71
FIGURE 49: REMOVAL OF TWO NUTS PRIOR TO INSTALLATION	71

FIGURE 50: ATTACHMENT TO OPTIONAL MOUNTING BRACKET	72
FIGURE 51: MID/ HIGH-GAIN WITH OPTIONAL MOUNTING BRACKET.....	72
FIGURE 52: OMNI WITH OPTIONAL MOUNTING BRACKET.....	73
FIGURE 53: SAFETY LANYARD ATTACHMENT.....	73
FIGURE 54: REMOVAL OF CONNECTOR FROM UNIT PRIOR TO INSTALLING CABLE	74
FIGURE 55: REMOVING RUBBER PLUG.....	74
FIGURE 56: INSERTION OF ETHERNET CABLE INTO CONNECTOR	75
FIGURE 57: CUTTING BACK OF CABLE JACKET.....	75
FIGURE 58: ALIGNMENT AND TRIMMING OF WIRES	76
FIGURE 59: EIA/TIA T568B	76
FIGURE 60: LOAD BAR DETAILS	77
FIGURE 61: INSERTING LOAD BAR INTO PLUG	77
FIGURE 62: MODULAR PLUG TERMINATION TOOL.....	77
FIGURE 63: ASSEMBLY OF RJ45 PLUG HOUSING	78
FIGURE 64: JACK AND PLUG ENGAGEMENT.....	78
FIGURE 65: GROUND WIRE ATTACHMENT.....	79

1. DOCUMENT OVERVIEW

This document describes the basic configuration and recommendations on physical installation of the Zebra Dart Vision Reader, which is part of the Zebra Dart UWB (Ultra Wide Band) product line. The document is provided as an installation and configuration guide for trained installers only.

2. PRODUCT DESCRIPTION AND FEATURES

The Dart Vision Reader provides robust asset visibility, including asset inventory and personnel access control applications. Utilizing patented short-pulse, UWB technology, the Dart Vision Reader offers a detection level that exceeds the capabilities of other Radio Frequency Identification (RFID) technologies. The Dart Vision Reader portfolio offers three standard integrated antenna (High-Gain, Mid-Gain and Omni) models to optimize the infrastructure required to achieve the robust Vision detection. Each Reader can simultaneously receive thousands of signals emitted by Zebra's active UWB Dart Tags with a read range upwards of 200 meters (650 ft) with the High-Gain model.

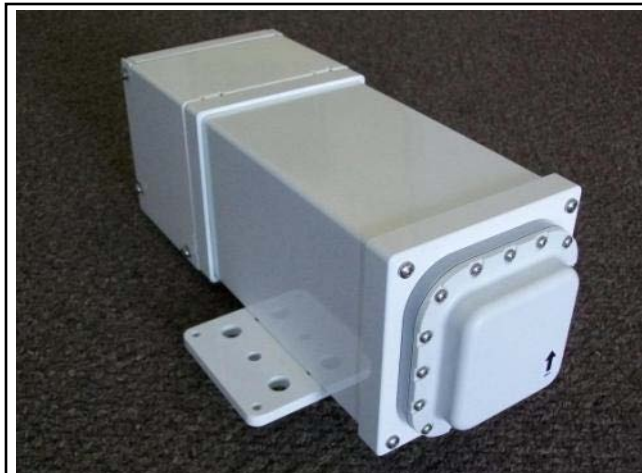


Figure 1: High-Gain Dart Vision Reader

In addition, the Zebra Dart Vision Reader is the world's first UWB Reader that is compatible with the new International UWB Standard, IEEE 802.15.4.f as well as the forthcoming release of ISO 24730-61. The Zebra Dart Tags are capable of operating in their native Zebra mode as well as ISO/IEEE mode. The Dart Vision Reader is able to simultaneously detect Zebra mode and ISO/IEEE mode tags. By adhering to International open standards; Zebra secures your infrastructure investment as it is compatible not only with Zebra's, but also other standards-compliant UWB tags.

The Dart Vision Reader can be used either as a stand-alone solution or as the perfect complement to the Dart UWB Real Time Locating System (RTLS). The Reader is fully weatherproof (IP67 rated) with an operational temperature range of -40°C to +55°C.

Combined with the Dart Tags ability to be detected in high multi-path, metallic environments, the Dart Vision Reader is able to handle the most robust, indoor and outdoor industrial environments.

The Dart Vision Reader is powered by standard IEEE 802.3af Power-over-Ethernet (PoE). The Reader supports the transport of the received Dart Tag messages via Ethernet or via its integral 802.11 b/g Wi-Fi client capability (Wi-Fi antennas sold separately). The Dart Vision Reader supports IPv4 and IPv6 network connectivity, allowing you optimal flexibility in migration to IPv6. The Dart Vision Reader offers the UWB Tag detection data in an easy to use format for further integration into RFID edge-ware solutions available from Zebra and a variety of vendors. See sections 6.1-6.4 for edgeware connection details.

The Dart Vision Reader utilizes standard 10/100 802.3 Ethernet connectivity via Cat-5 cables. It must be wired to a nearby IEEE 802.3af compliant Power over Ethernet (PoE) injector or network hub/switch that is PoE compliant, which is in turn connected to the network for connectivity to RFID edge-ware software. When operating in IPv4 mode, the Dart Vision Reader also supports standard DHCP IP address assignment, for ease of integration into your network environment. The maximum Ethernet cable run is 328 ft (100 m).

The Dart Vision Reader offers a Menu-driven management console, which is accessible via a Secure Shell connection for enhanced security. For diagnostic purposes, the Dart Vision Reader also supports a serial RS-232c connection to the management console.

The Dart Vision Reader offers unique features that greatly enhance the ability to implement a robust RFID solution. The Reader provides software enabled attenuation allowing the creation of localized Vision zones by dialing in the read range from inches to hundreds of feet with clear line of sight (LOS). A Dart Vision Reader also has the capability to be daisy-chained to another Vision Reader (and up to another two Readers with additional PoE injection) providing data via one Ethernet or Wi-Fi connection. This is a great advantage for those applications where multiple Readers will improve reading robustness or extend the range of Vision detection capability without the need to expand the IT network.

The Dart Vision Reader is the proven choice for exceptional visibility of critical assets and personnel in demanding industrial, manufacturing, supply chain and badging applications.

3. PRODUCT SPECIFICATIONS

Models	
Name	Part Number
Dart Vision Reader w/ High-Gain Antenna	UWD-1000-A-01AA
Dart Vision Reader w/ Mid-Gain Antenna	UWD-1000-A-02AA
Dart Vision Reader w/ Omni-Directional Antenna	UWD-1000-A-03AA
Performance	
Frequency Range (-10dB)	6.35 to 6.75 GHz
Antenna Gain	<ul style="list-style-type: none"> UWD-1000-A-01AA (High-Gain): 13.8 dBi UWD-1000-A-02AA (Mid-Gain): 6.5 dBi UWD-1000-A-03AA (Omni): 5.0 dBi
Environmental/Physical/Power	
Operating Temperature	-40°C to +55°C (-40°F to +131°F)
Environmental Sealing	IP67 (Dust & Direct Water Spray Protection)
Length (w/ Drip Shield)	<ul style="list-style-type: none"> Mid & High-Gain: 27.9 cm (11.0 in) Omni: 38.1 cm (15.0 in)
Width (w/ Mounting Bracket)	17.1 cm (6.75 in)
Height (w/ Mounting Bracket)	10.8 cm (4.25 in)
Weight	2.6 kg (5.8 lbs)
Power Consumption	Standard PoE 802.3af (37-57 VDC)
Power Supply	<ul style="list-style-type: none"> 802.11af PoE Injector, 100-240 VAC, Up to 100 m Ethernet cable Also commonly supplied by Customer 802.3af network switch, Up to 100 m Ethernet cable
Communications	
Ethernet	10/100 BaseT, CAT5 Jack, IEEE 802.3af compliant
Wi-Fi	<ul style="list-style-type: none"> IEEE 802.11 b/g, antenna ordered separately Transmit Frequency Band: 2400-2483.5 MHz Transmit Power: 20 dBm EIRP
Configuration & Diagnostics	<ul style="list-style-type: none"> Direct connection via Serial Port, RS-232, with adapter cable Telnet/ Secure Shell 2 via TCP/IP Ethernet interface
Regulatory Approvals	
<ul style="list-style-type: none"> FCC Part 15 subpart B & C; ICES-003, UL60950-1, UL60950-22; CAN/CSA-22.2 No. 60950-1-07, CAN/CSA-22.2 No. 60950-22 CE, EN 300 328, EN 301 489-17, EN 60950-1 	

Accessories	
Name	Part Number
Wi-Fi Antenna – Indoor/Outdoor Omni-Directional 5.2dBi	AK-151-00
Wi-Fi Antenna – Indoor/Outdoor Directional, 13.5dBi	AK-153-00
Wi-Fi Antenna – Indoor Omni-Directional 2.2dBi	AK-170-00
IEEE 802.3af Power over Ethernet (PoE) Injector	EP-025-00
Mounting Arm, Dart Vision Reader	UM-120-00
Console Cable for Dart Vision Reader	CBL-440-00

Specifications subject to change without notice

4. INSTALLATION AND MOUNTING

The Dart UWB Reader can be connected and powered by the customer network via an 802.3af PoE Ethernet Switch or Wi-Fi network in conjunction with an 802.3af PoE mid-span injector. See the accessory list for part numbers of both the available PoE injector and Wi-Fi Antenna. Follow all safety warnings and cautions.



Warning - Electrical Shock: A protective earthing conductor with green and yellow insulation, minimum of 18 AWG, shall be installed to the protective earthing terminal of the metal enclosure. National electrical codes shall be followed to install facility protective earthing conductor to the protective earthing terminal.



Warning - Electrical Shock: No operator serviceable parts inside. Refer servicing to qualified personnel. To prevent electrical shock, do not remove covers.



Caution - The Dart Reader hardware must be installed by a qualified service technician.



Caution – For outdoor installation of the Dart Reader, only PoE devices rated for powering outdoor devices are allowed.

4.1 Items Required (supplied by installer)

1. Cable, CAT5e, non-shielded

a. Stranded Wire

Cable Type: 8 positions

Conductor size: 24 AWG

Conductor type: 7 strand copper

Contact insulator diameter: 0.99 mm maximum

RJ45 plug accepts cable outer diameter range: 4.83 mm ~ 6.73 mm

b. Solid Wire

Cable Type: 8 positions

Conductor size: 24 AWG

Conductor type: copper

Contact insulator diameter: 0.99 mm maximum

RJ45 plug accepts loose, pliable cable outer diameter range: 4.83 mm ~ 6.73 mm

RJ45 plug accepts hard, rigid cable outer diameter range: 4.83 mm ~ 6.73 mm

2. Wire, hook-up, 12 AWG, stranded, Green

4.2 Tools Required /Suggested

1. Phillips screwdriver, #2

2. 7/16 wrench or socket

3. 9/16 wrench or socket

4. Fixed Hook Spanner wrench, for 1-3/16 to 1-17/64 (30 -32mm) circle diameter


Can be obtained at McMaster Carr. McMaster Carr Part Number: 6975A14


5. Modular Plug Termination Tool, 8P8C. Conec Part Number: 360X30029X


4.3 Connections

There are 3 basic connections for the Dart Vision Reader; 2 Ethernet RJ-45's and 1 RP-TNC for Wi-Fi antenna. There is also a lug/terminal provided for earth ground connection.

1. **Ethernet Port A:** Is the primary Ethernet and Power port for the unit(s) via PoE.
2. **Ethernet Port B:** Is the secondary and proprietary port, which has 2 primary functions.
 - a. First it is used to daisy chain a 2nd Dart Vision Reader. Second it can be used as a Serial Diagnostic port via the use of a special cable. (Cable, RJ45 to DB9) Zebra part CBL-440-00

 **Caution** – Only use the Zebra approved Console Cable CBL-440-00, to directly connect to a serial port connection. The use of a non-approved Console Cable to Port B will damage the unit and potentially any computer equipment connected to it. Any additional cable or extension may cause serial port errors and the unit not to boot properly.

 **Caution** – When daisy chaining (see section 4.4), be sure to turn off power to all PoE units in a chain before making the daisy chain connections. Failure to do so will result in damage to the units.

–  **Caution** – Port B is intended only for cascade/daisy chain or Console connection only, never apply power to Port B with any PoE device. Damage will occur to the unit.

3. The **Wi-Fi** Antenna Port which is a RP-TNC.

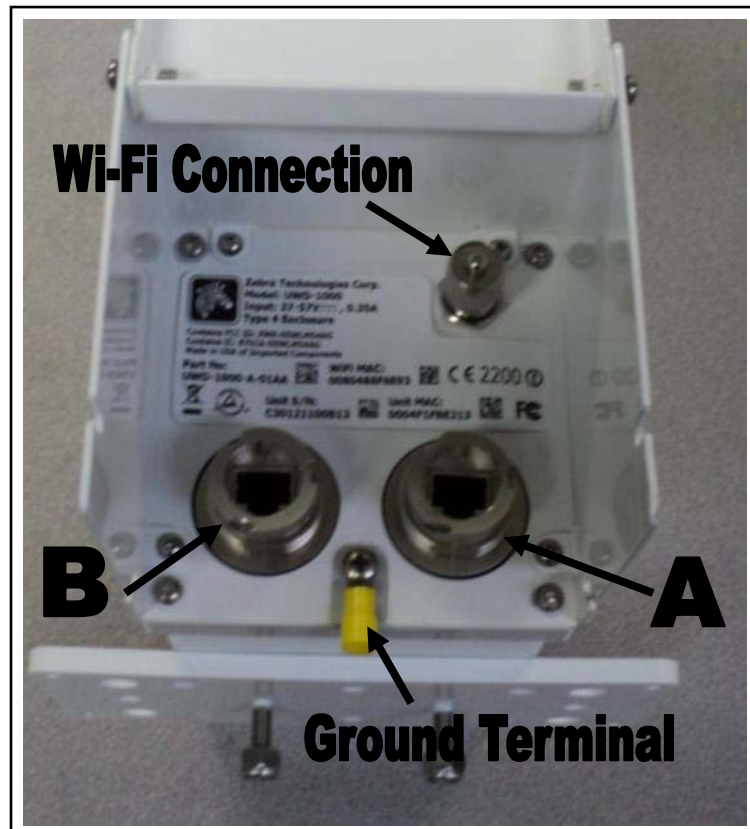


Figure 2: DVR Connections (*Note: RJ-45 Back Shells removed in Figure*)

5. CONFIGURATION

The Dart Vision Reader is pre-configured with the network connection to be DHCP IP address assignment. The Management Console is accessible via SSH and, for diagnostic purposes, serial RS-232c (via accessory cable). Either connection method provides the user with a menu driven management console. The menu driven system allows the end user to configure a static IP address if desired, RF sensitivity, and also allows for configuration of the Wireless LAN, along with other functions. See important note below on the use of Telnet to access the management console.

5.1 Configuration Connection

Once the unit is powered up, the user can then configure other settings via the menu system. The menu system on the Dart Vision Reader is a numbered menu system with the number being entered for a desired menu or setting that is to be modified. After a configuration change is made, the user is generally prompted to hit the escape key 'ESC' and enter a password (ff2) to save the settings. Multiple settings can be changed in one session, and for the settings to take effect, the unit needs to be reset. To establish an SSH or Telnet menu session, the user must know the IP address that the DHCP server issued or the host name assigned to the Dart Vision Reader and then establish a standard network connection to the DVR through Port A. For a serial HyperTerminal session, connect to Port B of the DVR with the Zebra DB-9 to RJ-45 serial cable.

Note


SSH is the preferred method of communicating with the Dart Vision Reader. Use the HyperTerminal is for initial set-up before being installed on a network for setting static IP addresses, if DHCP is not used. In addition to SSH, Dart Vision Reader also supports Telnet, and is disabled by default. Note that Telnet sends user name and passwords across the network in the clear. So it is recommended that Telnet be disabled after the Dart Vision Reader configuration is completed.

5.1.1 Serial Port Communication

Using the serial port accessory cable (Console Cable CBL-440-00) that connects to the Port-B RJ-45, the Dart Vision Reader may be configured using HyperTerminal or equivalent. The serial port is intended for use for diagnostics purposes. The Dart Vision Reader should not be permanently connected to serial cables. In particular, when using the serial connection, allow the unit to boot up before connecting the serial cable. The menu structure is the same and all the same settings are available.

Communication Parameters:

- Accessory cable DB-9 to RJ45
19200 baud, RS-232c 8 data bits, 1 stop bit, no parity, no hardware flow control

 **Caution** – Only use the Zebra approved Console Cable CBL-440-00, to directly connect to a serial port connection. The use of a non-approved Console Cable to Port B will damage the unit. Any additional cable or extension may cause serial port errors and the unit not to boot properly.

5.1.2 Establishing an SSH Connection

The Dart Vision Reader supports Secure Shell (SSH), SSH2 being preferred over SSH1. Either the IP address or host name of the unit must be known in order to establish an SSH session.

An SSH session may be established using a freely available application called OpenSSH (supplied by your Zebra Professional Services staff or visit <http://openssh.com> for further details).

To establish the SSH session, you must use Zebra-supplied key, called “ssh_zebra_rsa_key” and the “blowfish” passphrase.

The following section shows how to use the OpenSSH application.

The provided zip archive contains all you need to use OpenSSH. Unzip the file and put it anywhere on your hard drive.

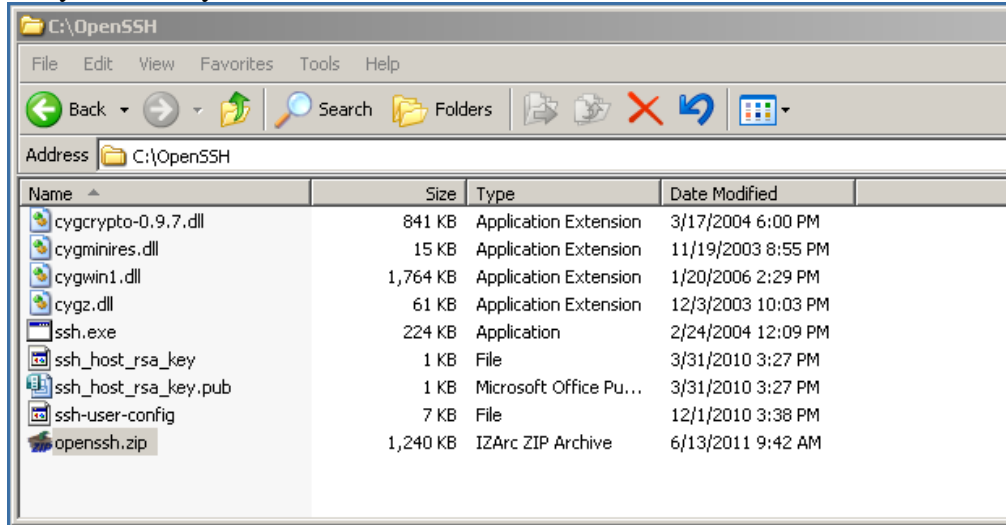
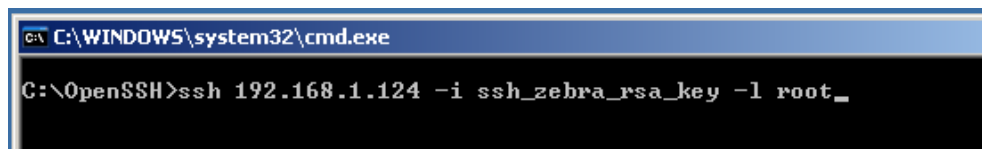


Figure 5: SSH Files

1. Launch a command prompt, and change directory to the folder where you placed the executable ssh.exe file, for example, "C:\OpenSSH".

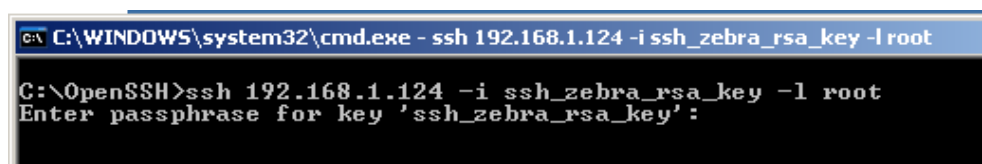


2. Enter the following command at the command prompt:
'ssh <IP-address> -i ssh_zebra_rsa_key -l root'



Note: The first time you do this from most computers you will be told "can't establish...." And it will ask you "are you sure?" ...enter 'yes'

3. The application will prompt you for a passphrase, use: 'blowfish'



4. Dart Vision Reader then presents secure access to its Configuration menu.

```

C:\WINDOWS\system32\cmd.exe - ssh 192.168.1.124 -i ssh_zebra_rsa_key -l root
Enter passphrase for key 'ssh_zebra_rsa_key':

BusyBox v1.2.1 (2011.02.14-15:44+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

07:02                                     Build: Jun 8 2011 08:54:17

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                        IP=192.168.1.124

Main Menu:
1. Ethernet Settings
2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>

```

Figure 6: SSH Main Menu

5.2 Unit Configuration

There are several options that can be configured by the user in the Dart Vision Reader.

- Ethernet Settings (Wired LAN and Wireless LAN)
 - DNS (Domain Name Server)
 - SNMP (Simple Network Management Protocol)
 - Telnet ON/OFF option
- 802.11 Client Setting
- Firmware Upgrade
- Manufacturing information and Setting.
- Read Range (Tag Reception Sensitivity)
- ISO/IEEE Modes (UWB Tag mode).

The following sections will give examples of most of the common configuration options. It is advised that the end user work with a member of Professional Services to select the appropriate configuration for the application.

The Dart Vision firmware version V5.0.0 (Build Date July 10 2012) and later includes support for the SNMP and DNS network protocols. The 'Ethernet Settings' menu in the sensor menu includes menus to enable and configure these protocols. The DNS settings are located under both the LAN (wired interface) and WLAN (wireless interface) respectively. The SNMP is its own menu under the 'Ethernet Settings' menu, shown in

example below. Each of the settings can be change, with similar procedures as all other menus.

```

C:\SSH\ssh.exe
1. Boot Method          FLASH
2. LAN
3. WLAN
4. IP Forwarding        OFF
5. SNMP
6. Telnet               ON
>reading NVRam keys...
2:50:17                               Build: Jun 12 2012 11:38:42

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                               IP=192.168.5.50

SNMP Settings Menu:
1. System Name:         zbr-fbe207
2. System Location:     2858 De La Cruz Blvd, Santa Clara, CA 95050
3. Contact Information: 800-508-5326
4. ReadCommunity:       public
5. WriteCommunity:      private
6. Trap Destination:    Disabled
7. Trap Community:      public
8. Agent:               ON
>

```

5.2.1 Disabling Telnet

While SSH (Secure Shell) is one of the most secure methods of communication to the Dart Vision Reader, the option for standard Telnet communication has been left as an option. The unit ships with the Telnet Option disabled by default, but may be enabled as follows.

Select 1, Ethernet settings, from the Main menu. Then the Dart Vision Reader application will display the following sub-menu:

```

C:\WINDOWS\system32\cmd.exe - ssh 192.168.1.124 -i ssh_zebra_rsa_key -l root
2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>reading NVRam keys...
29:48                               Build: Jun  8 2011 08:54:17

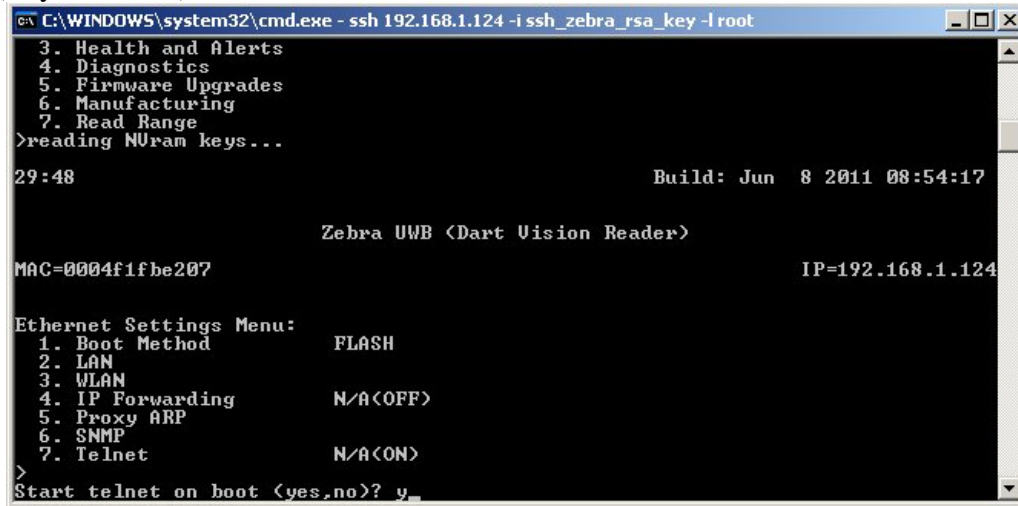
                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                               IP=192.168.1.124

Ethernet Settings Menu:
1. Boot Method          FLASH
2. LAN
3. WLAN
4. IP Forwarding        N/A<OFF>
5. Proxy ARP
6. SNMP
7. Telnet               N/A<ON>
>

```

Next, select numbered item that corresponds to Telnet, simply respond with n to disable (or y to enable) Telnet.



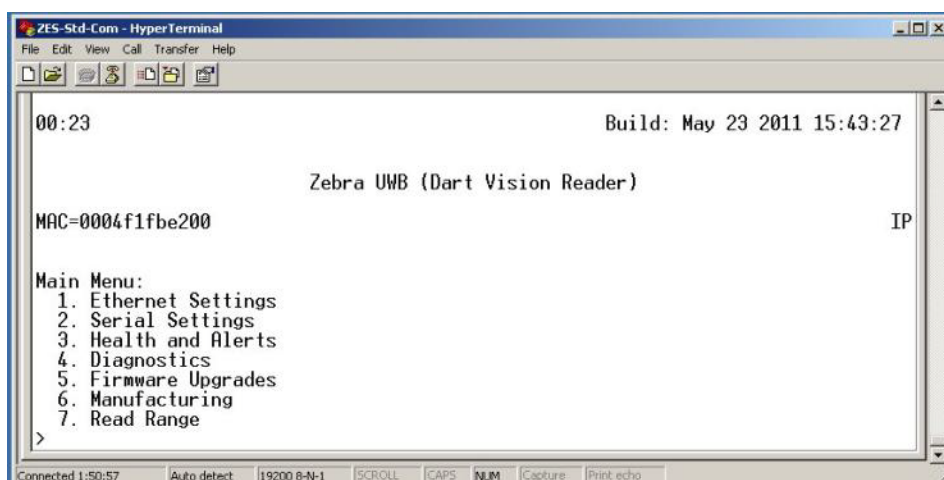
```
C:\WINDOWS\system32\cmd.exe - ssh 192.168.1.124 -i ssh_zebra_rsa_key -l root
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>reading NVRam keys...
29:48 Build: Jun  8 2011 08:54:17
Zebra UWB <Dart Vision Reader>
MAC=0004f1fbe207 IP=192.168.1.124
Ethernet Settings Menu:
1. Boot Method FLASH
2. LAN
3. WLAN
4. IP Forwarding N/A<OFF>
5. Proxy ARP
6. SNMP
7. Telnet N/A<ON>
>
Start telnet on boot <yes,no>? y_
```

Press the ESC key and enter the “ff2” password when prompted. The unit will update NVRAM, save your changes, and ask if you wish to reset the unit. After reset, the unit will boot without Telnet support.

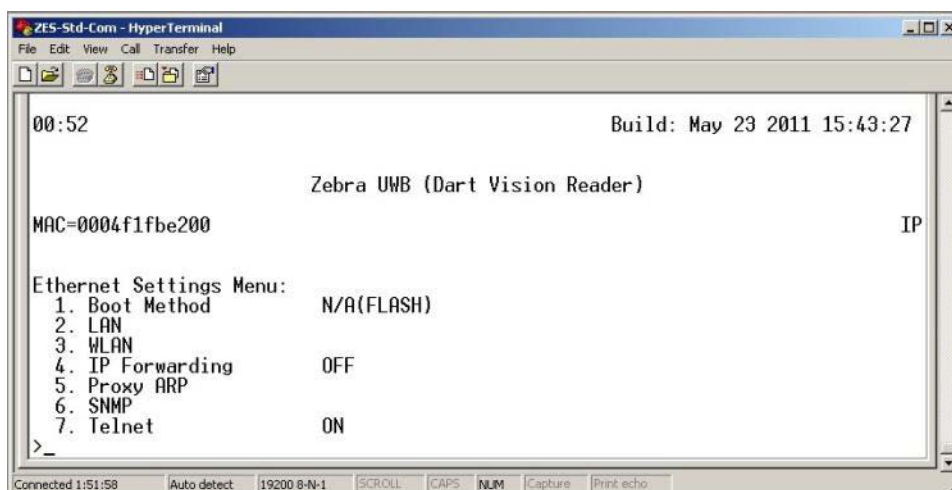
5.2.2 Wired LAN Setting IP Address/DSN

The Dart Vision Reader is generally used in DHCP networking environments, with automatic population of DNS entries via DHCP. For some installations or demo purposes, the unit can be set for a Static IP address and/or manual DNS configuration per the requirement of the local network. Note, however, that manual DNS configuration is overridden by DHCP. The following example will go through the menu system using the Serial connection to set-up/change a static IP address via the Reader secondary Ethernet port (Port B) with the use of a special RJ45 to DB9 cable (Zebra P/N CBL-440-00). Note the DNS menu will allow for the setting of 'Host Name', 'Domain Name', 'Primary DNS' and Alternate 'DNS' through a similar process as setting the static IP address. These following instruction and menus are would be the same if the user is going through Telnet or SSH connection.

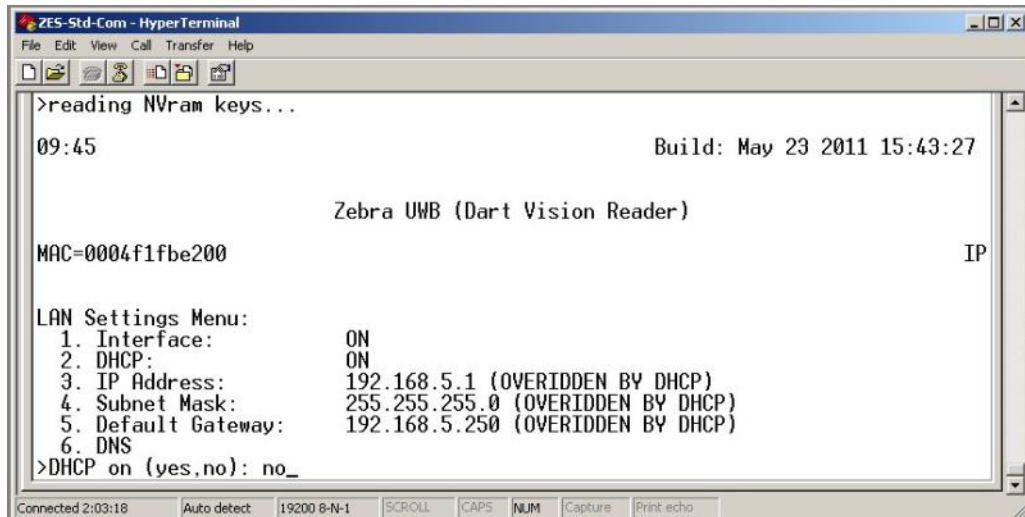
From the Main Menu Select (1) for Ethernet Settings



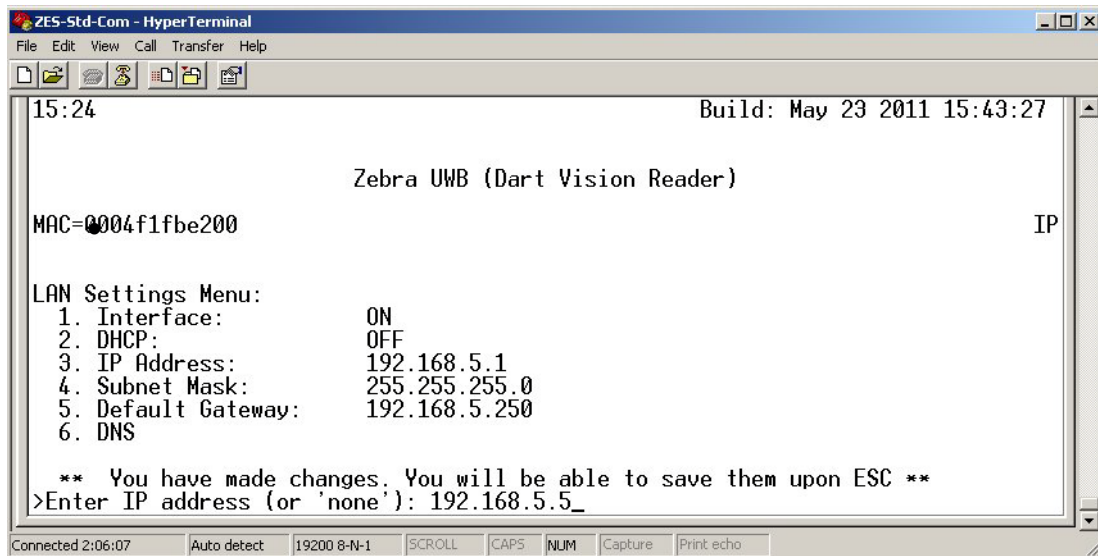
From the Ethernet Menu Select (2) LAN



From the LAN Menu Select (2) DHCP, "no" turns DHCP off

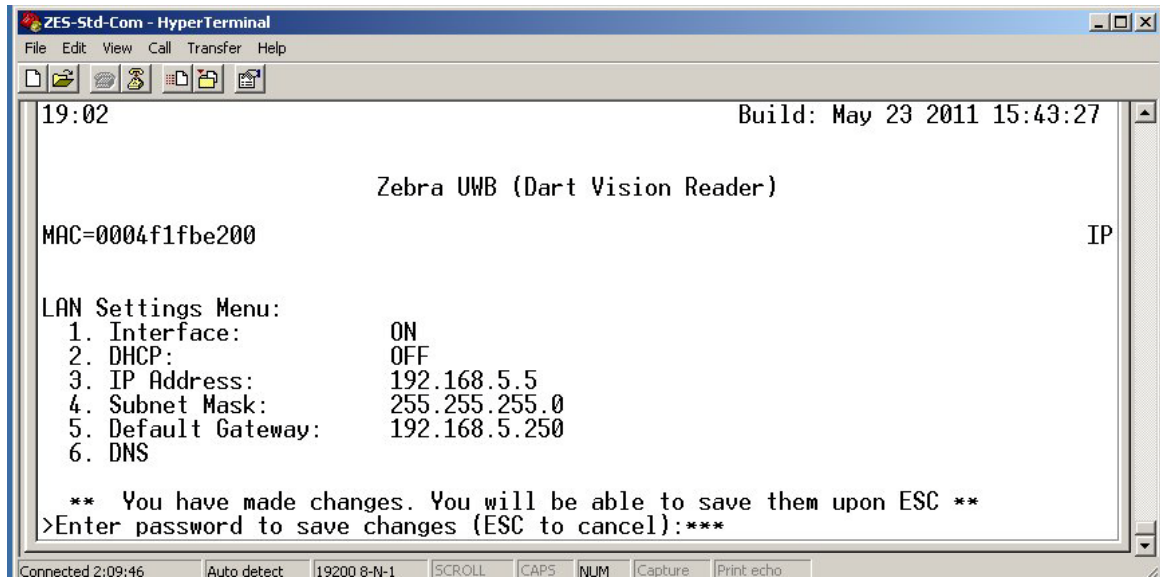


From LAN Menu select (3) to set a static IP address, enter desired IP address and press Enter/Return.



Repeat these steps for settings Subnet Mask and Default Gateway.

Once completed settings everything, press escape (Esc). The unit will ask for a password, which is (ff2) and Enter/Return.



After the unit has saved the settings, it will ask for to Reset the unit, and enter (yes) Enter/Return. Note the unit requires a Reset to the changes to take effect.

5.3 LAN Wi-Fi Client Configuration

The Wi-Fi client card is a separate network interface embedded in the Dart Vision Reader. There are several configuration options that must be configured for the wireless client to function properly.

Under the WLAN Menu

- First Interface must be “ON” (Required)
- IP address must be set or DHCP must be enabled.
 - Subnet Mask
 - Default Gateway (optional)
 - DNS Server (optional)

Under the Manufacturing Menu

- Country Code (Required)

Under Diagnostics >> 802.11 Client Menu

- Wifi Antenna Type should be verified the default is ANT4941 (Must be confirmed)
- WPA Supplicant must be enabled (Required ON)

The first step in enabling the wireless LAN client is as follows.

From the Main Menu, select 1. Ethernet Settings

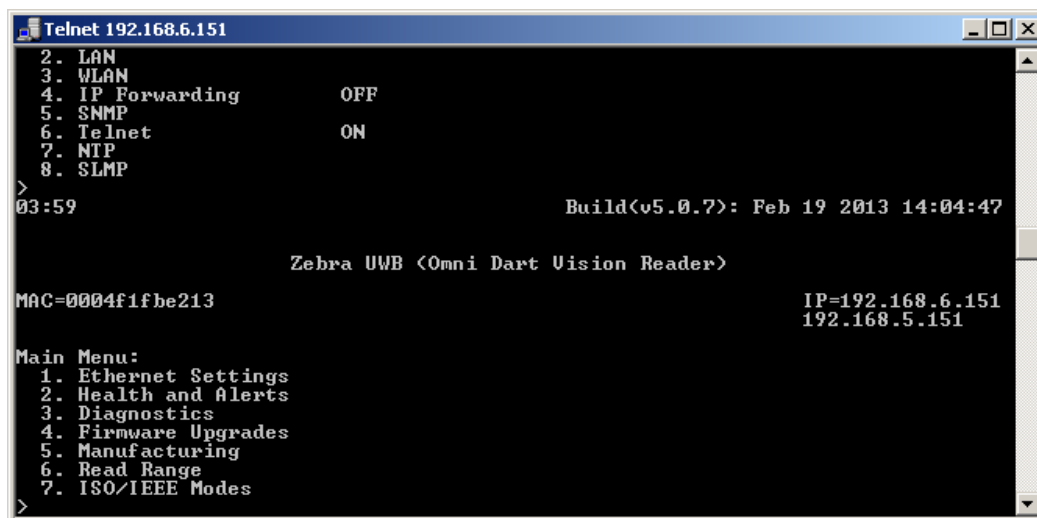


Figure 7: Main Menu

Under Ethernet Menu, -> 3. WLAN



```

C:\OpenSSH\ssh.exe
1. Boot Method      FLASH
2. LAN
3. WLAN
4. IP Forwarding    N/A<OFF>
5. Proxy ARP
6. SNMP            N/A<ON>
7. Telnet
>reading NVRam keys...

42:30                               Build: Jun  8 2011 08:54:17

                               Zebra UWB <Dart Vision Reader>

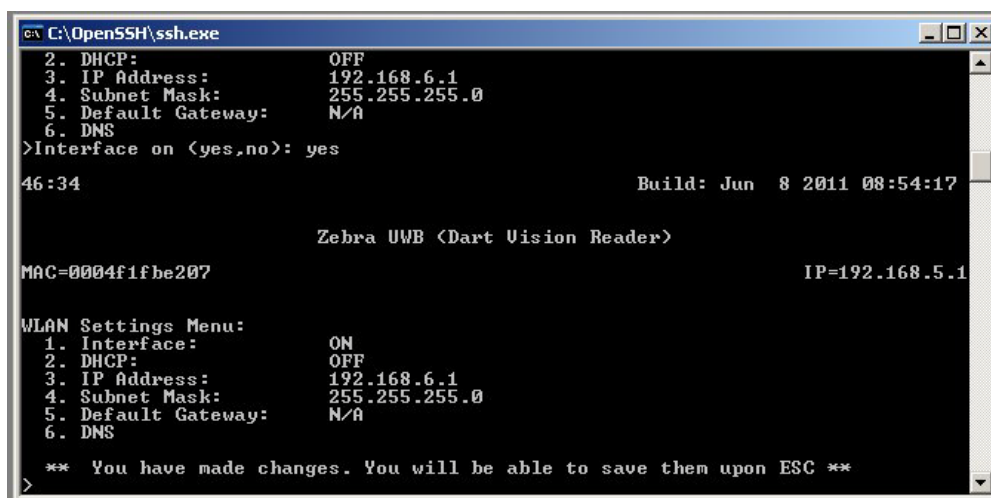
MAC=0004f1fbe207                               IP=192.168.5.1

WLAN Settings Menu:
1. Interface:      OFF
2. DHCP:           OFF
3. IP Address:     192.168.6.1
4. Subnet Mask:    255.255.255.0
5. Default Gateway: N/A
6. DNS
>

```

Figure 8: WLAN Menu

Set 1. Interface to ON, then set IP/DHCP settings, subnet mask, default gateway, and DNS as appropriate.



```

C:\OpenSSH\ssh.exe
2. DHCP:           OFF
3. IP Address:     192.168.6.1
4. Subnet Mask:    255.255.255.0
5. Default Gateway: N/A
6. DNS
>Interface on <yes,no>: yes

46:34                               Build: Jun  8 2011 08:54:17

                               Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                               IP=192.168.5.1

WLAN Settings Menu:
1. Interface:      ON
2. DHCP:           OFF
3. IP Address:     192.168.6.1
4. Subnet Mask:    255.255.255.0
5. Default Gateway: N/A
6. DNS
>
** You have made changes. You will be able to save them upon ESC **

```

Figure 9: WLAN ON

Save the configuration changes by using ESC key entering the password “ff2”, note changes take effect on unit reboot/reset.

Note

Although the Dart Vision Reader has two network interfaces (its Ethernet and Wi-Fi ports) in the vast majority of cases, only its Wi-Fi port is actually connected to an operational network. If enabled, the Ethernet port must be configured to be on a different subnet than the Wi-Fi port. In this case, it is recommended that a non-routable IP address be used such as 192.168.5.0/255.255.255.0, to enable local diagnostics via the Ethernet port.

5.3.1 Set Country Code of Operation

The Country Code of Operation is required to be set by a Zebra trained Project Manager, Product Support Personnel, or Partner. The Country code setting required to meet RF (Radio Frequency) Compliance standards of the selected country. This setting must be set for the Wi-Fi Client to operate on the Dart Vision Reader



Important: See FSB-325_DVR_Wi-Fi_CountryCode or Appendix G before upgrading a UWD-1000-A-0xxA with active Wi-Fi Client.

These settings can be applied via either a Secure Shell or Telnet connection over an Ethernet network or direct via a Null mode serial cable with a HyperTerminal session. The Menu Systems are the same. The examples below are from an Ethernet Secure Shell connection.

- Starting from the main Menu, select the number that corresponds with “Manufacturing”

```

C:\SSH\ssh.exe
Main Menu:
1. Ethernet Settings
2. Health and Alerts
3. Diagnostics
4. Firmware Upgrades
5. Manufacturing
6. Read Range
7. ISO/IEEE Modes
>
03:00 Build: Jun 12 2012 11:38:42

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207 IP=192.168.5.50

Main Menu:
1. Ethernet Settings
2. Health and Alerts
3. Diagnostics
4. Firmware Upgrades
5. Manufacturing
6. Read Range
7. ISO/IEEE Modes
>

```

Figure 10: Main Menu Country Code

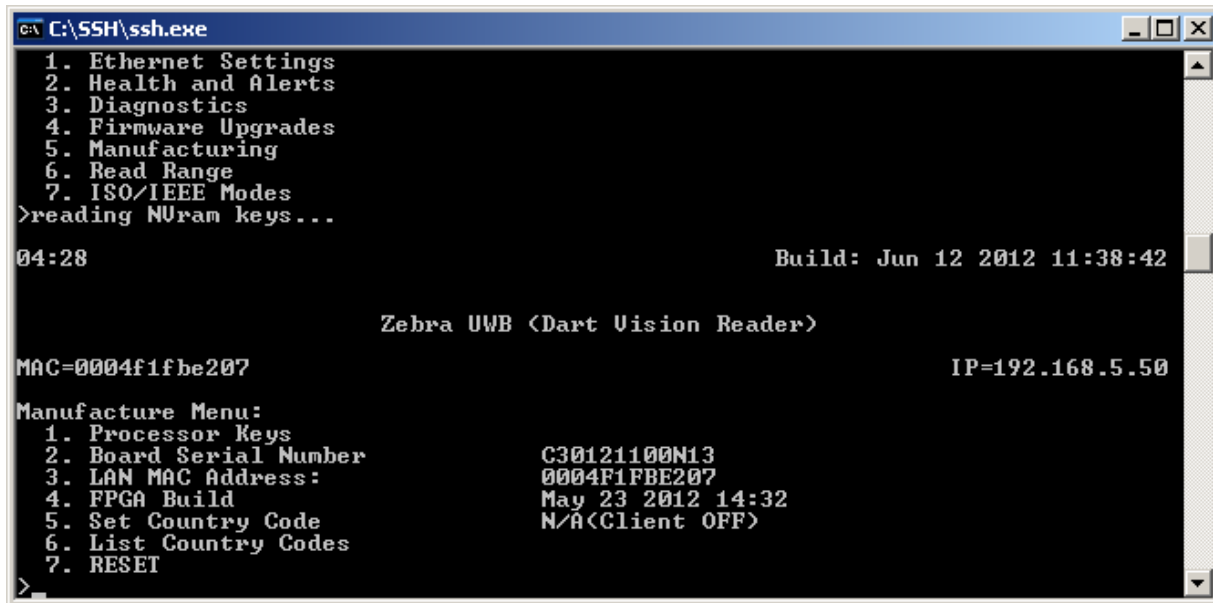


Figure 11: Manufacturing Menu

Select item number that corresponds to “List Country Code”. Take note of appropriate country code.

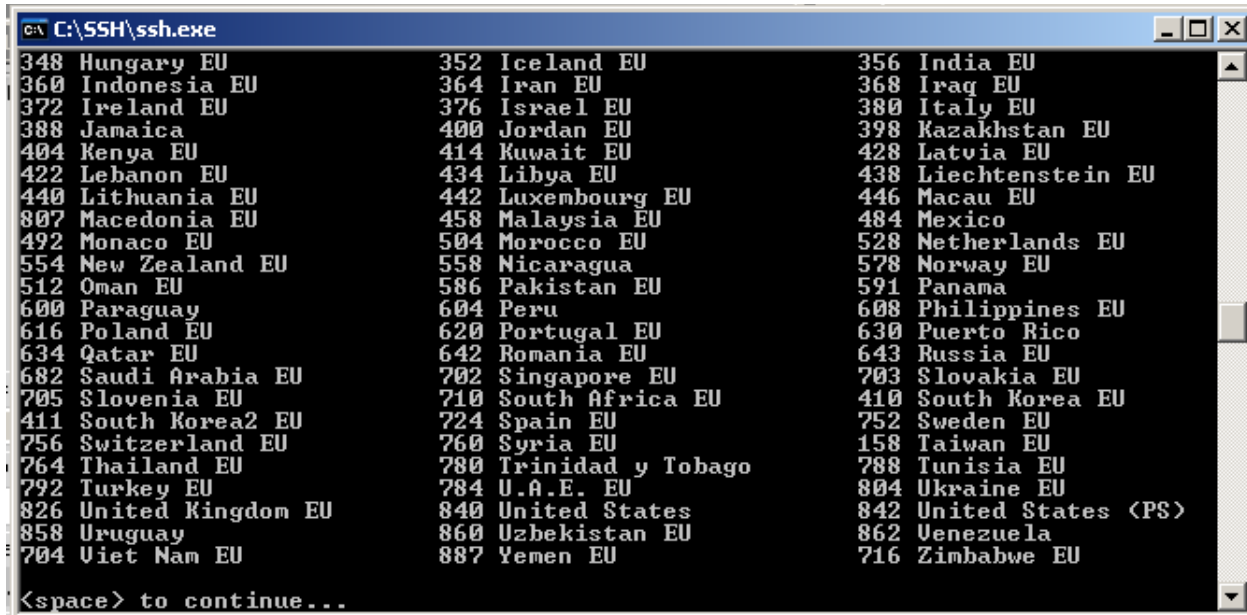
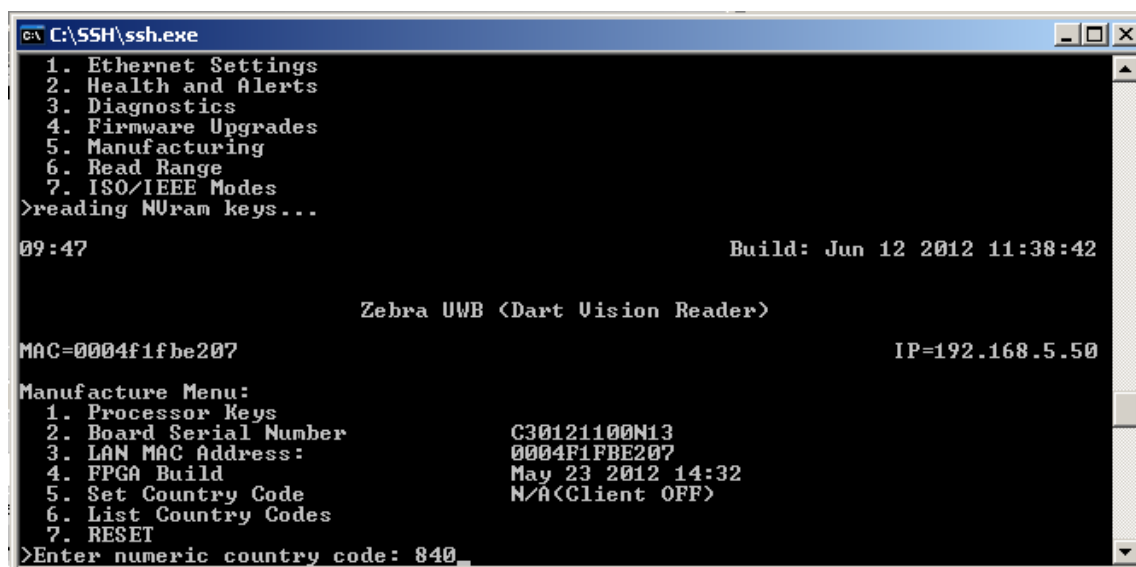


Figure 12: Country Code List

- Then select the number that corresponds with “Set Country Code”, and enter the appropriate Country Code number, which can also be found in [Appendix D: Dart Vision Reader Country Code Key Setting](#).



```

C:\SSH\ssh.exe
1. Ethernet Settings
2. Health and Alerts
3. Diagnostics
4. Firmware Upgrades
5. Manufacturing
6. Read Range
7. ISO/IEEE Modes
>reading NURam keys...

09:47                                     Build: Jun 12 2012 11:38:42

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                                     IP=192.168.5.50

Manufacture Menu:
1. Processor Keys
2. Board Serial Number
3. LAN MAC Address:
4. FPGA Build
5. Set Country Code
6. List Country Codes
7. RESET
>Enter numeric country code: 840_

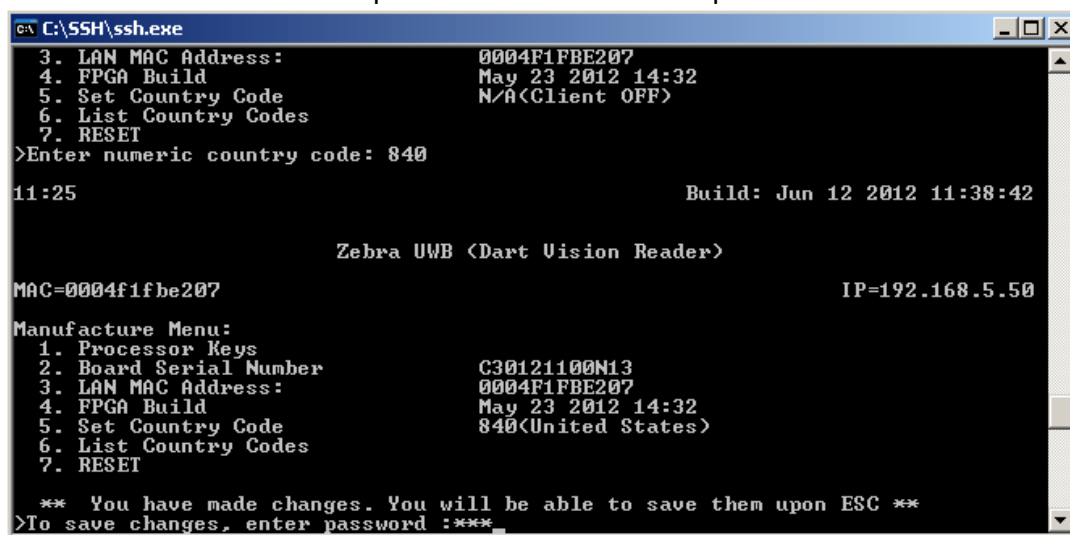
```

Figure 13: Set Country Code



Important: [Appendix D: Dart Vision Reader Country Code Key Setting](#) contains the list of Countries that were approved at the time of release of this User's Guide, which corresponds to version V 5.0.0 of the DVR firmware. For the most up to date list of approved Country Codes and settings files, please refer to FSB-326_DVR_Country Code List document.

- Now Escape and enter “ff2” for the password.



```

C:\SSH\ssh.exe
3. LAN MAC Address:
4. FPGA Build
5. Set Country Code
6. List Country Codes
7. RESET
>Enter numeric country code: 840

11:25                                     Build: Jun 12 2012 11:38:42

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                                     IP=192.168.5.50

Manufacture Menu:
1. Processor Keys
2. Board Serial Number
3. LAN MAC Address:
4. FPGA Build
5. Set Country Code
6. List Country Codes
7. RESET
** You have made changes. You will be able to save them upon ESC **
>To save changes, enter password :***_

```

Figure 14: Save Settings**5.3.2 Setting Wi-Fi Antenna Type**

The Wi-Fi antenna setting is available in firmware versions V5.0.0 or later (Build Date July 10 2012 or later). The antenna selection is set under 802.11Client Menu, which is reached from the Main Menu via the Diagnostics Menu. There are currently three antenna choices to select from, and these match the Wi-Fi antennas that are offered on the marketing price list. The selection is entered under the “Select Antenna” menu item.

Antenna Options:

- AK-170-00 2.2dBi Dipole Antenna (Rubber Duck) , Selection **ANT4941**
- AK-151-00 5.2dBi Omni Antenna, Selection **ANT2506**
- AK-153-00 13.5 Yagi, Directional Mast Mount Antenna, **ANT1949**

From the *Main Menu*, select *Diagnostics*, and from the *Diagnostics Menu* select *802.11 Client*. Then, in the 802.11 Client Menu, select the number that corresponds to “Select Antenna”, and enter the value that corresponds to the antenna that will be attached to the Dart Vision Reader.

```

C:\SSH\ssh.exe
MAC=0004f1fhe207 IP=192.168.5.50
802.11 Client Menu:
1. Interface Status
2. Set DataRate (Mb/s)          AUTO
3. Set TxPower (dBm)           20
4. Set Roaming Threshold       8
5. Set RTS Threshold           OFF
6. Set Fragmentation Threshold OFF
7. Set Max Retries             ?
8. Select Antenna              ANT4941
9. AP Scan
a. Encryption
b. Certificate Dates
c. WPA Supplicant              OFF
d. WPA Supplicant Status
e. WPA Supplicant Script
f. WPA Supplicant Reconfigure
g. WPA Supplicant Reassociate
>Supported antennas:
Cisco ANT4941 Indoor Omni 2.2dBi (Zebra AK-170-00)
Cisco ANT2506 Outdoor Omni 5.2dBi (Zebra AK-151-00)
Cisco ANT1949 Yagi 13.5dBi (Zebra AK-153-00)
Select antenna (ANT4941, ANT2506, or ANT1949): ANT4941
  
```

Figure 15: Antenna Type

Escape “ESC” to save selection, ‘ff2’ is the password. The unit will need to be rebooted for the setting to take effect.

5.3.3 Configuring the Wi-Fi Network Parameters

All the Wi-Fi network parameters are under the control of a file called `wpa_supplicant.conf`. In order to configure the Dart Vision Reader Wi-Fi client card to access your network, you must first edit or create this file to match your Wi-Fi network configuration and then upload the file onto the Dart Vision Reader unit.

Create your `wpa_supplicant.conf` file with your Linux-friendly editor and place a copy of it on the RTLS Server's `ftproot` directory structure.

Uploading the `wpa_supplicant.conf` file onto the Dart Vision Reader.

From the Main Menu-> Firmware Upgrade Menu.

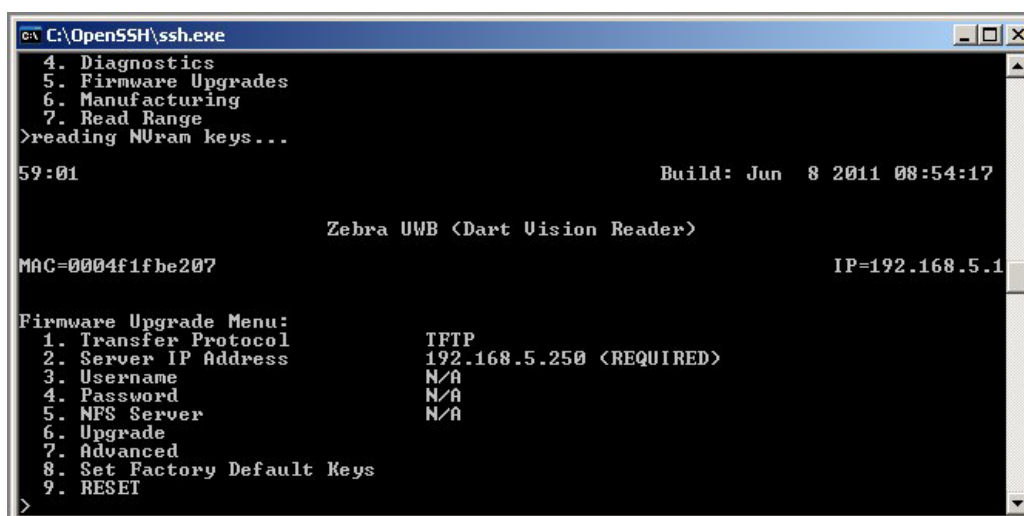
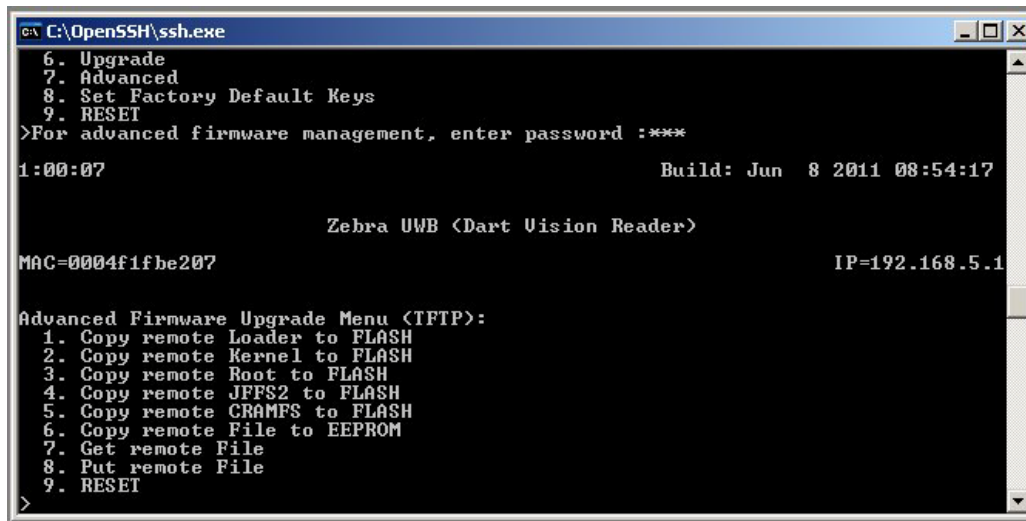


Figure 16: File Upload

Select the proper Transfer Protocol for the file transfer, valid options are TFTP, FTP, SFTP, and NFS. Set IP address of the server where the `wpa_supplicant` file is located. Last, set User and Password if applicable.

Now, select item, Advanced, and enter the ff2 password. The Reader will present the following menu.



```

C:\OpenSSH\ssh.exe
6. Upgrade
7. Advanced
8. Set Factory Default Keys
9. RESET
>For advanced firmware management, enter password :***
1:00:07 Build: Jun 8 2011 08:54:17

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbc207 IP=192.168.5.1

Advanced Firmware Upgrade Menu <IFTP>:
1. Copy remote Loader to FLASH
2. Copy remote Kernel to FLASH
3. Copy remote Root to FLASH
4. Copy remote JFFS2 to FLASH
5. Copy remote CRAMFS to FLASH
6. Copy remote File to EEPROM
7. Get remote File
8. Put remote File
9. RESET
>

```

Figure 17: Advance Menu

Now, select, Get remote file. The Dart Vision Reader asks for the path to the remote file. As you have placed the file in the server's root directly, simply enter 'wpa_supplicant.conf'. Now, the unit asks for a local path, i.e., where the file should be placed in the Dart Vision Reader itself. See section 2 of this document for details about configuring a wpa_supplicant.conf file.

Enter '/mnt/jffs2/wpa_supplicant.conf'. Be sure to enter the path and file names correctly, as an error here will prevent normal Wi-Fi operations. Dart Vision Reader asks you to confirm before the file is uploaded. Accept by entering y. When the operation completes, your screen should look like this:

```

C:\SSH\ssh.exe
03:07 Build: Jun 12 2012 11:38:42

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207 IP=192.168.5.50
192.168.6.50

Advanced Firmware Upgrade Menu <TFTP>:
1. Copy remote Loader to FLASH
2. Copy remote Kernel to FLASH
3. Copy remote Root to FLASH
4. Copy remote JFFS2 to FLASH
5. Copy remote CRAMFS to FLASH
6. Copy remote File to EEPROM
7. Get remote File
8. Put remote File
9. RESET
>Enter path to remote file: wpa_supplicant.conf
Enter local path: /mnt/jffs2/wpa_supplicant.conf
About to copy, are you sure? <yes, no>:yes
Copying <wpa_supplicant.conf> to </mnt/jffs2/wpa_supplicant.conf>...
Operation complete.
<space> to continue...

```

Figure 18: Get Remote File

Now, that the wpa_supplicant.conf file has been uploaded, you must enable wpa_supplicant processing on the Dart Vision Reader itself.

From the top menu, select item number that corresponds to “Diagnostics”.

```

C:\SSH\ssh.exe
05:52 Build: Jun 12 2012 11:38:42

Zebra UWB <Dart Vision Reader>

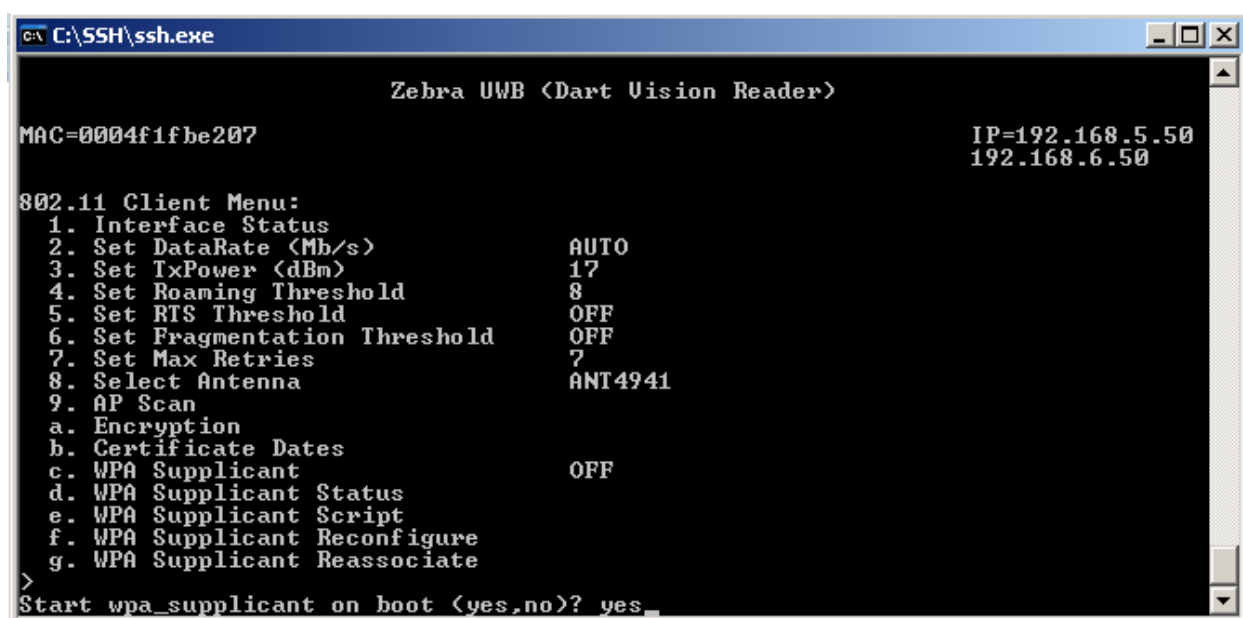
MAC=0004f1fbe207 IP=192.168.5.50
192.168.6.50

Diagnostics Menu:
1. Uptime
2. Ping
3. Flash LED's
4. Memory
5. Drives
6. Network Interfaces
7. Network Connections
8. 802.11 Client
9. Route Table
a. Task List
b. Task CPU Usage
c. CPU Monitor
d. NFS
e. Watchdog
f. RESET
>

```

Figure 19: 802.11 Client Menu

Select item number that corresponds to “802.11 client”.



```

C:\SSH\ssh.exe

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                                IP=192.168.5.50
                                                192.168.6.50

802.11 Client Menu:
1. Interface Status
2. Set DataRate <Mb/s>                AUTO
3. Set TxPower <dBm>                  17
4. Set Roaming Threshold              8
5. Set RTS Threshold                  OFF
6. Set Fragmentation Threshold        OFF
7. Set Max Retries                    7
8. Select Antenna                     ANT4941
9. AP Scan
a. Encryption
b. Certificate Dates
c. WPA Supplicant                     OFF
d. WPA Supplicant Status
e. WPA Supplicant Script
f. WPA Supplicant Reconfigure
g. WPA Supplicant Reassociate
>
Start wpa_supplicant on boot <yes,no>? yes_

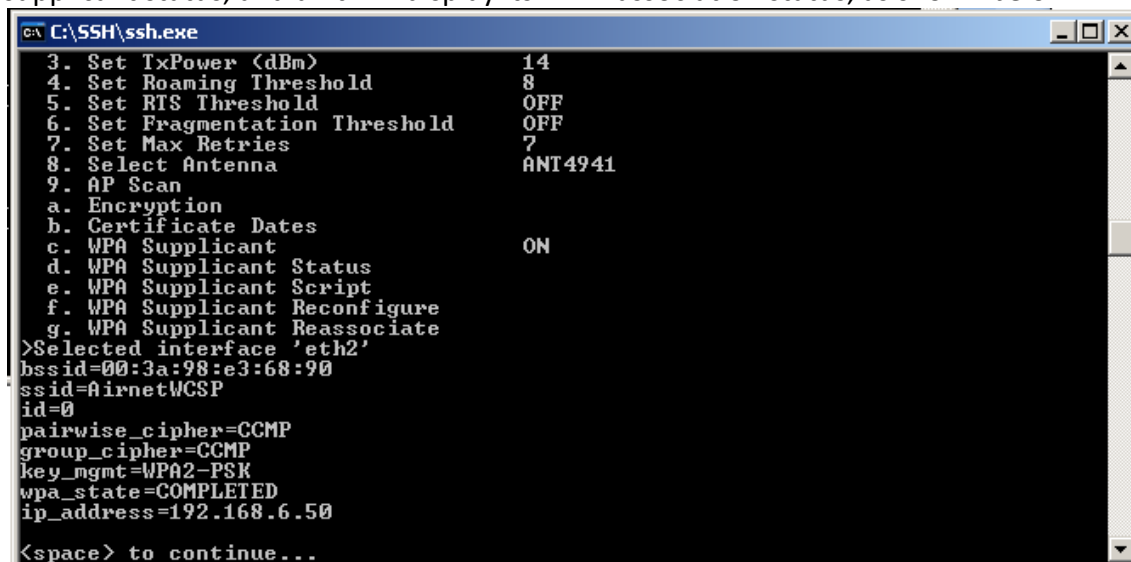
```

Figure 20: WPA_Supplicant at Boot

And now, select item “WPA Supplicant”, to turn on (yes) wpa_supplicant processing. Press ESC to save changes, and enter the ff2 password, and space to continue.

The unit must be reset in order for wpa_supplicant processing to take effect.

To check the unit’s Wi-Fi network connection, select Diagnostics, from the Main Menu, and then select 802.11 Client, from the Diagnostics menu. Now, select d, WPA supplicant status, and unit will display its Wi-Fi association status, as shown below.



```

C:\SSH\ssh.exe

3. Set TxPower <dBm>                14
4. Set Roaming Threshold              8
5. Set RTS Threshold                  OFF
6. Set Fragmentation Threshold        OFF
7. Set Max Retries                    7
8. Select Antenna                     ANT4941
9. AP Scan
a. Encryption
b. Certificate Dates
c. WPA Supplicant                     ON
d. WPA Supplicant Status
e. WPA Supplicant Script
f. WPA Supplicant Reconfigure
g. WPA Supplicant Reassociate
>Selected interface 'eth2'
bssid=00:3a:98:e3:68:90
ssid=AirnetWCSP
id=0
pairwise_cipher=CCMP
group_cipher=CCMP
key_mgmt=WPA2-PSK
wpa_state=COMPLETED
ip_address=192.168.6.50
<space> to continue...

```

Figure 21: WPA Confirmation

6. SUPPLICANT CONFIGURATIONS

For the following are examples of wpa_supplicant configurations files for various security modes. Note the 'wpa_supplicant.conf' must be edited with text edit that is compatible with Unix based editors. **Do not** use NotePad to edit the wpa_supplicant.conf file. See latter sections for details on how to download/upload the wpa_supplicant to and from the Dart Vision Reader. The 'wpa_supplicant.conf' file is located in the '/mnt/jffs2' directory.

6.1 Personal Configurations

The following three configuration modes are for Personal (PSK) configurations, in other words a Radius Server is not required.

6.1.1 WEP configuration

WEP (Wired Equivalent Privacy) security is the most basic of security, and not widely used as it is relatively easy to compromise. **Does not require a Radius Server.** The following network block is an example of a wpa_supplicant.conf configured to run WEP security. The settings are contained in what is referred to as a "Network Block", the text between the open and closed parentheses after 'network='. A comment is designated by a pound/number sign '#', comments are not part of the settings, just placed as notes.

```
# WEP
network={
    ssid="NGLS"
    key_mgmt=NONE
    wep_key0="wherenet12345"
    wep_key1="wherenet12345"
    wep_key2="wherenet12345"
    wep_key3="wherenet12345"
    wep_tx_keyidx=0
    auth_alg=OPEN
}
```

Comment noting this is for WEP security.

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

key_mgmt: list of accepted authenticated key management protocols

- NONE = WPA is not used; plaintext or static WEP could be used

wep_key0..3: Static WEP key (ASCII in double quotation, e.g. "abcde" or

- # hex without quotation, e.g., 0102030405)
- # wep_tx_keyidx: Default WEP key index (TX) (0..3)

wep_tx_keyidx: The active WEP key index (TX) (0..3)

auth_alg: list of allowed IEEE 802.11 authentication algorithms

- OPEN = Open System authentication (required for WPA/WPA2)
- SHARED = Shared Key authentication (requires static WEP keys)

6.1.2 WPA-PSK network block in “wpa_supplicant.conf”

The following configuration example of is for WPA (Wifi Protected Access), which is stronger than WEP encryption but not as strong as WPA2. WPA is TKIP and WPA2 is CCMP.

```
# WPA-PSK
network={
    ssid="NGLS"
    key_mgmt=WPA-PSK
    pairwise=TKIP
    group=TKIP
    psk="ZebraKey"
}
```

Comment noting this is for WPA Personal security. (PSK)

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

key_mgmt: list of accepted authenticated key management protocols

- WPA-PSK = WPA pre-shared key (this requires 'psk' field)

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0, WPA
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40
-

psk: WPA preshared key; 256-bit pre-shared key

- The key used in WPA-PSK mode can be entered either as 64 hex-digits, i.e., 32 bytes or as an ASCII passphrase (in which case, the real PSK will be generated using the passphrase and SSID). ASCII passphrase must be between 8 and 63 characters (inclusive). ext:<name of external PSK field> format can be used to indicate that the PSK/passphrase is stored in external storage.
- This field is not needed, if WPA-EAP is used.

6.1.3 WPA2-PSK network block in “wpa_supplicant.conf”

The following configuration example of is for WPA2 (Wifi Protected Access version), which is stronger then WPA encryption and the most secure of the personal (PSK) security modes.

```
# WPA2-PSK
network={
    ssid="NGLS"
    key_mgmt=WPA-PSK
    pairwise=CCMP
    group=CCMP
    psk="ZebraKey"
}
```

Comment noting this is for WPA2 Personal security version 2. (PSK)

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

key_mgmt: list of accepted authenticated key management protocols

- WPA-PSK = WPA pre-shared key (this requires 'psk' field)

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40

psk: WPA preshared key; 256-bit pre-shared key

- The key used in WPA-PSK mode can be entered either as 64 hex-digits, i.e., 32 bytes or as an ASCII passphrase (in which case, the real PSK will be generated using the passphrase and SSID). ASCII passphrase must be between 8 and 63 characters (inclusive). ext:<name of external PSK field> format can be used to indicate that the PSK/passphrase is stored in external storage.
- This field is not needed, if WPA-EAP is used.

6.2 Extensible Authentication Protocol (EAP)

The following section will give examples of the most common Extensible Authentication Protocol (EAP) methods. These methods typically require a Radius Server to be running, and are typically used in enterprise applications. Some of the supported security methods will also require you to copy over digital certificate files (*.pem format).

- Dart Vision Reader support Authentication Methods
 - EAP-TLS
 - EAP-PEAP/MSCHAPv2 (both PEAPv0 and PEAPv1)
 - EAP-PEAP/TLS (both PEAPv0 and PEAPv1)
 - EAP-PEAP/GTC (both PEAPv0 and PEAPv1)
 - EAP-PEAP/OTP (both PEAPv0 and PEAPv1)
 - EAP-PEAP/MD5-Challenge (both PEAPv0 and PEAPv1)
 - EAP-TTLS/EAP-MD5-Challenge
 - EAP-TTLS/EAP-GTC
 - EAP-TTLS/EAP-OTP
 - EAP-TTLS/EAP-MSCHAPv2
 - EAP-TTLS/EAP-TLS
 - EAP-TTLS/MSCHAPv2
 - EAP-TTLS/MSCHAP
 - EAP-TTLS/PAP
 - EAP-TTLS/CHAP
 - EAP-SIM
 - EAP-AKA
 - EAP-PSK
 - EAP-PAX
 - LEAP
 - (following methods are supported, but since they do not generate keying material, they cannot be used with WPA or IEEE 802.1X WEP keying)
 - EAP-MD5-Challenge
 - EAP-MSCHAPv2
 - EAP-GTC

- Basic steps to authentication, only 802.1X traffic until RADIUS-ACCEPT sent to AP

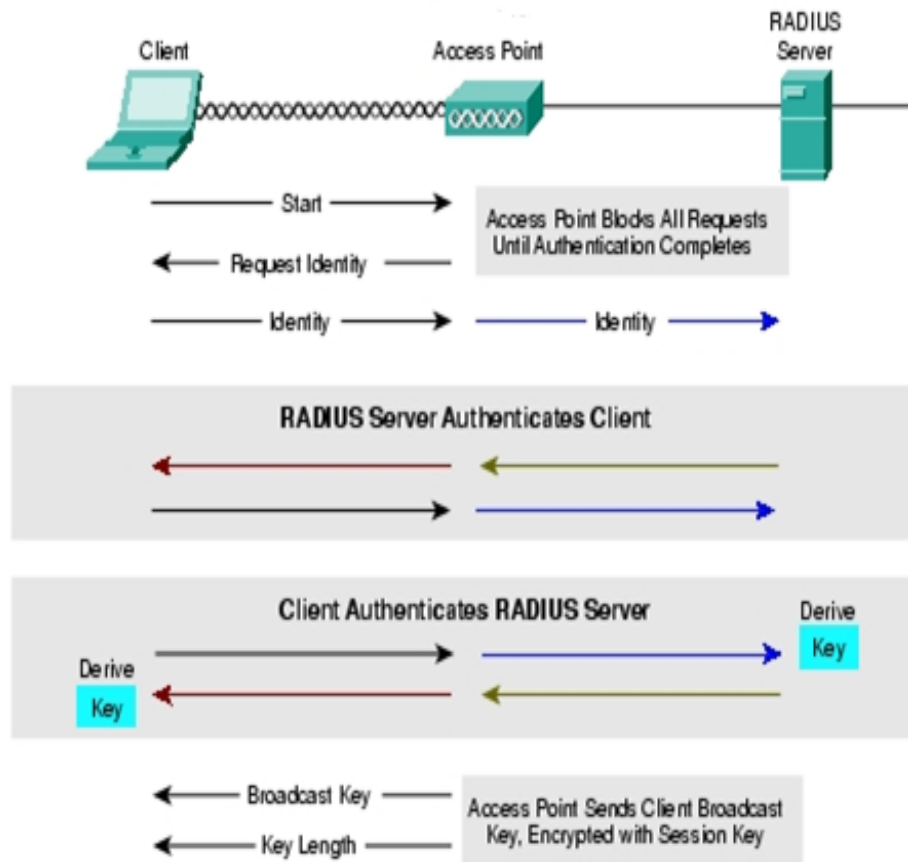


Figure 22: EAP Authentication

6.3 Digital Certificates

- Strong EAP types require client and server to use certificates for verification
 - Methods are either “Client and Server” or “Server Only”
- Certificates in general are issued and signed by a Certificate Authority
 - Validates the identity of a certificate holder
 - (CA) Signs the certificate to confirm that it has not been altered
 - Example certificate are preloaded on the Dart Vision Reader in the ‘/mnt/jffs2/certs/’ directory.
 - The certificates can be download/uploaded via the advanced menu in firmware upgrades in a similar method to the wpa_supplicant.conf file.
- Contents include but not limited to:
 - Version, Serial Number, Activation and/or expiration date, Public Key Algorithm, “Subject” Public Key, Certificate Signature Algorithm, Certificate Signature
 - The Dart Vision Reader, menu system uses “openssl” to display certificate dates
 - User can choose to initialize a runtime clock to validate certificate(s) prior to framework time synchronization
 - Certificate “subject” must match roll during authentication
- Supported filename extension are .pem and .der
 - Example/test certificates preloaded on the Dart Vision Reader are .pem format
 - /mnt/jffs2/certs/ca.pem and /mnt/jffs2/certs/client.pem

Example of the certificate upload. From the Main Menu->Firmware Upgrades -> Advanced -> Get Remote File

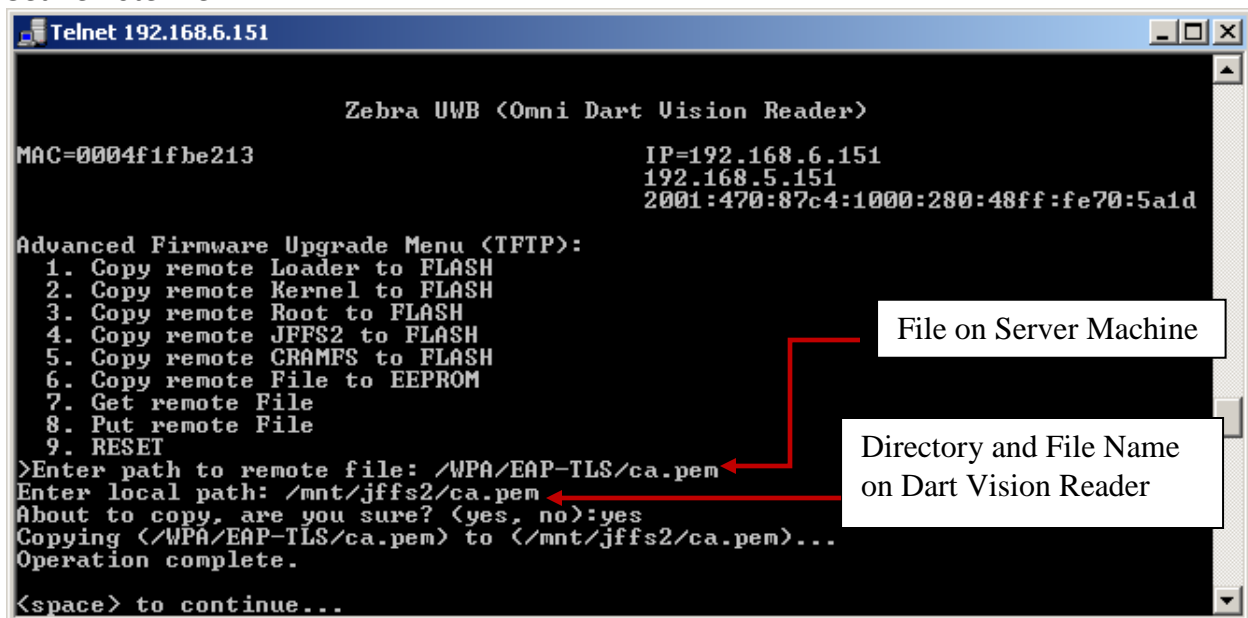
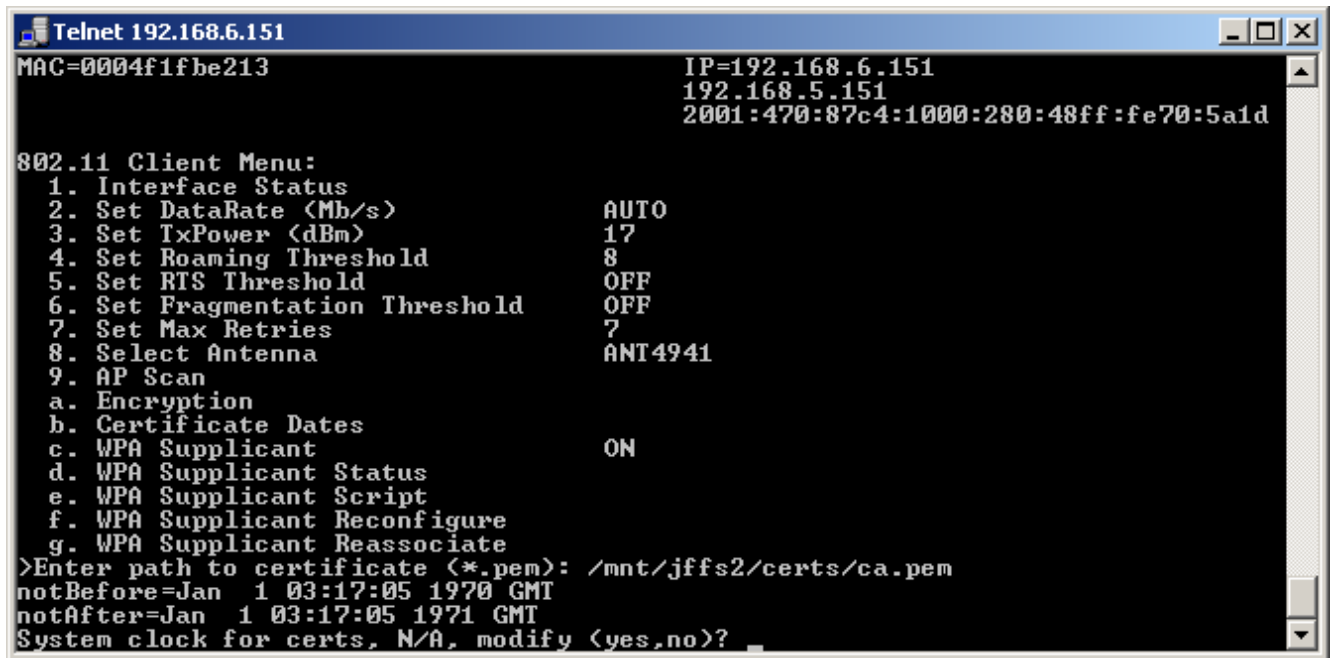


Figure 23: Certificate Upload

After the server Authentication File is upload, the server certificate dates need to be entered into the Dart Vision Reader menu System. Depending of the firmware load the Default start date of the Dart Vision Reader most likely will not fall within the dates that the Certificate Authority is valid for. The Dart Vision Reader can read and list the valid dates in the Certificate Authority (.pem) file. For the Wifi Client to associate the Certificates Dates must to set correctly in the operating system of the Dart Vision Reader. Once the unit receives it's site data and connection to the Blink Service then it receives the date and time of the server that it is connected to.

For example the 'ca.pem' that was loaded has a valid date range from



```

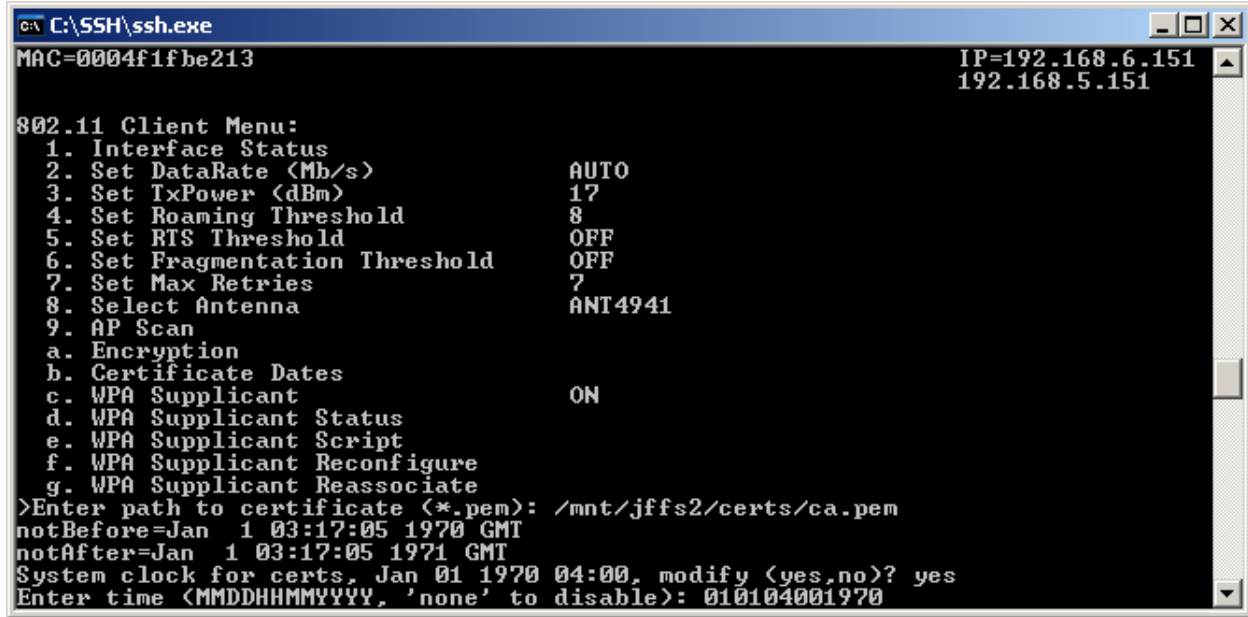
Telnet 192.168.6.151
MAC=0004f1fbe213                               IP=192.168.6.151
                                                192.168.5.151
                                                2001:470:87c4:1000:280:48ff:fe70:5a1d

802.11 Client Menu:
 1. Interface Status
 2. Set DataRate (Mb/s)          AUTO
 3. Set TxPower (dBm)           17
 4. Set Roaming Threshold       8
 5. Set RTS Threshold           OFF
 6. Set Fragmentation Threshold OFF
 7. Set Max Retries             7
 8. Select Antenna              ANT4941
 9. AP Scan
 a. Encryption
 b. Certificate Dates
 c. WPA Supplicant              ON
 d. WPA Supplicant Status
 e. WPA Supplicant Script
 f. WPA Supplicant Reconfigure
 g. WPA Supplicant Reassociate
>Enter path to certificate (*.pem): /mnt/jffs2/certs/ca.pem
notBefore=Jan  1 03:17:05 1970 GMT
notAfter=Jan  1 03:17:05 1971 GMT
System clock for certs, N/A, modify <yes,no>?

```

Figure 24: Certificate Data

Set valid date in Month Day Hour Minute Year format (MMDDHHMMYYYY) with two digits for Month, two digits for Days, two digits for Hours, two digits for Minutes, and four digits for Year.



```

C:\SSH\ssh.exe
MAC=0004f1fbe213 IP=192.168.6.151
192.168.5.151

802.11 Client Menu:
1. Interface Status
2. Set DataRate (Mb/s) AUTO
3. Set TxPower (dBm) 17
4. Set Roaming Threshold 8
5. Set RTS Threshold OFF
6. Set Fragmentation Threshold OFF
7. Set Max Retries 7
8. Select Antenna ANT4941
9. AP Scan
  a. Encryption
  b. Certificate Dates
  c. WPA Supplicant ON
  d. WPA Supplicant Status
  e. WPA Supplicant Script
  f. WPA Supplicant Reconfigure
  g. WPA Supplicant Reassociate
>Enter path to certificate (*.pem): /mnt/jffs2/certs/ca.pem
notBefore=Jan 1 03:17:05 1970 GMT
notAfter=Jan 1 03:17:05 1971 GMT
System clock for certs, Jan 01 1970 04:00, modify <yes,no>? yes
Enter time (MMDDHHMMYYYY, 'none' to disable): 010104001970
  
```

Figure 25: Set Valid Date

After setting certificate valid date, escape 'ecs' to save settings. (Password 'ff2'). Then reboot the unit.

6.4 Common EAP Methods

The following examples are of more common EAP methods for the Dart Vision Reader. It is highly recommended that professional services work with the customers IT department to test the used security method at site before installation of equipment. All these methods are configured through the "wpa_supplicant.conf" file, see section 1.1.3 for wpa_supplicant upload.

- EAP-TLS (Transport Layer Security)
 - Uses PKI to secure communications with RADIUS authentication server
 - Provides mutual authentication of client-to-server and server-to-client
 - Both client and server must be assigned a digital certificate signed by a Certificate Authority (CA)
 - Key exchange to establish dynamic WEP or TKIP keys
- EAP-TTLS (Tunneled TLS EAP)
 - Server-to-client authentication requires a certificate, client-to-server uses the established secure tunnel
 - Removes burden of issuing client certificates
 - Client-to-server authentication can use existing authentication protocol and infrastructure
 - TTLS server can act as a proxy between an AP and a legacy RADIUS server
- EAP-PEAP (Protected EAP)
 - Very similar to TTLS, no client certificate required
 - Uses TLS only to authenticate server-to-client but not client-to-server
 - Client and server exchange EAP messages encapsulated within TLS messages
 - TLS messages are authenticated and encrypted using TLS session keys
 - Many flavors, PEAPv0/EAP-MSCHAPv2 is by far the most common
- EAP-LEAP (Lightweight EAP)
 - Cisco proprietary
 - Weak security, breakable in minutes
 - Essentially an enhanced version of EAP-MD5 with dynamic key rotation and mutual authentication
 - MD5 was never meant to be used on an un-trusted wireless network
 - Relies solely on MS-CHAPv2 (Microsoft Challenge Handshake Authentication Protocol) to protect user credentials
 - Sends usernames in cleat-text

6.4.1 WPA (Enterprise) EAP-TLS

The following network block is an example of EAP-TLS and requires a Radius Server.

```
# EAP-TLS ←
network={
    ssid="NGLS"
    scan_ssid=1
    key_mgmt=WPA-EAP
    pairwise=TKIP
    group=TKIP
    eap=TLS
    eapol_flags=0
    identity="root"
    password="password"
    ca_cert="/mnt/jffs2/certs/ca.pem"
    client_cert="/mnt/jffs2/certs/client.pem"
    private_key="/mnt/jffs2/certs/client.key"
    private_key_passwd="whatever"
    fragment_size=1024
}
```

Comment noting this is for EAP-TLS

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

scan_ssid:

- 0 = do not scan this SSID with specific Probe Request frames (default)
- 1 = scan with SSID-specific Probe Request frames (this can be used to find APs that do not accept broadcast SSID or use multiple SSIDs; this will add latency to scanning, so enable this only when needed)

key_mgmt: list of accepted authenticated key management protocols

- WPA-EAP = WPA using EAP authentication
- IEEE8021X = IEEE 802.1X using EAP authentication and (optionally) dynamically generated.

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40

eap: Pre-configured EAP method (TLS, TTLS, PEAP, or LEAP).

This optional field can be used to specify which EAP method will be used with this credential. If not set, the EAP method is selected automatically based on ANQP information (e.g., NAI Realm).

Continued on next Page

Continuation of EAP-TLS wpa_supplicant network block

eapol_flags: IEEE 802.1X/EAPOL options (bit field)

- Dynamic WEP key require for non-WPA mode
- bit0 (1): require dynamically generated unicast WEP key
- bit1 (2): require dynamically generated broadcast WEP key
- (3 = require both keys; default)

identity: Identity string for EAP, must be in quotes.

password: Password string for EAP, must be in quotes.

ca_cert: File path to CA certificate file.

- This file can have one or more trusted CA certificates.
- If ca_cert is not included, server certificate will not be verified. This is insecure and the CA file should always be configured.

client_cert: File path to client certificate file (PEM/DER)

private_key_passwd: Password for private key file, must be in quotes.

fragment_size: Maximum EAP fragment size in bytes (default 1398).

- This value limits the fragment size for EAP methods that support fragmentation (e.g., EAP-TLS and EAP-PEAP). This value should be set small enough to make the EAP messages fit in MTU of the network interface used for EAPOL. The default value is suitable for most cases.

6.4.2 WPA2 (Enterprise) EAP-TTLS

The following is an example of an EAP-TTLS network block, a Radius server is required.

```
# EAP-TTLS ←
network={
    ssid="NGLS"
    scan_ssid=1
    key_mgmt=WPA-EAP
    pairwise=CCMP TKIP
    group=CCMP TKIP
    eap=TTLS
    eapol_flags=0
    identity="root"
    password="password"
    anonymous_identity="anonymous"
    ca_cert="/mnt/jffs2/certs/ca.pem"
    phase2="auth=MD5"
}
```

Comment noting this is for EAP-TTLS

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

scan_ssid:

- 0 = do not scan this SSID with specific Probe Request frames (default)
- 1 = scan with SSID-specific Probe Request frames (this can be used to find APs that do not accept broadcast SSID or use multiple SSIDs; this will add latency to scanning, so enable this only when needed)

key_mgmt: list of accepted authenticated key management protocols

- WPA-EAP = WPA using EAP authentication
- IEEE8021X = IEEE 802.1X using EAP authentication and (optionally) dynamically generated.

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40

eap: Pre-configured EAP method (TLS, TTLS, PEAP, or LEAP).

This optional field can be used to specify which EAP method will be used with this credential. If not set, the EAP method is selected automatically based on ANQP information (e.g., NAI Realm).

Continued on next Page

EAP-TTLS network block explanation continued.

eapol_flags: IEEE 802.1X/EAPOL options (bit field)

- Dynamic WEP key require for non-WPA mode
- bit0 (1): require dynamically generated unicast WEP key
- bit1 (2): require dynamically generated broadcast WEP key
- (3 = require both keys; default)

identity: Identity string for EAP, must be in quotes.

password: Password string for EAP, must be in quotes.

anonymous_identity: Anonymous identity string for EAP (to be used as the unencrypted identity with EAP types that support different tunnelled identity, e.g., EAP-TTLS). This field can also be

ca_cert: File path to CA certificate file.

- This file can have one or more trusted CA certificates.
- If ca_cert is not included, server certificate will not be verified. This is insecure and the CA file should always be configured.

phase2: inner authentication with TLS tunnel) parameters (tring with field-value pairs, e.g., "auth=MSCHAPV2" for EAP-PEAP or

- "autheap=MSCHAPV2 autheap=MD5" for EAP-TTLS)

6.4.3 WPA2 (Enterprise) EAP-PEAP

The following is an example of an EAP-PEAP network block, a Radius Server is required.

```
# EAP-PEAP ←
network={
    ssid="NGLS"
    scan_ssid=1
    key_mgmt=WPA-EAP
    pairwise=CCMP TKIP
    group=CCMP TKIP
    eap=PEAP
    eapol_flags=0
    identity="root"
    password="password"
    ca_cert="/mnt/jffs2/certs/ca.pem"
    phase1="peapver=0"
    phase2="MSCHAPV2"
}
```

Comment noting this is for EAP-PEAP

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

scan_ssid:

- 0 = do not scan this SSID with specific Probe Request frames (default)
- 1 = scan with SSID-specific Probe Request frames (this can be used to find APs that do not accept broadcast SSID or use multiple SSIDs; this will add latency to scanning, so enable this only when needed)

key_mgmt: list of accepted authenticated key management protocols

- WPA-EAP = WPA using EAP authentication
- IEEE8021X = IEEE 802.1X using EAP authentication and (optionally) dynamically generated.

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40

eap: Pre-configured EAP method (TLS, TTLS, PEAP, or LEAP).

This optional field can be used to specify which EAP method will be used with this credential. If not set, the EAP method is selected automatically based on ANQP information (e.g., NAI Realm).

Continued on next Page

This is a continuation of the EAP-PEAP network block.

eapol_flags: IEEE 802.1X/EAPOL options (bit field)

- Dynamic WEP key require for non-WPA mode
- bit0 (1): require dynamically generated unicast WEP key
- bit1 (2): require dynamically generated broadcast WEP key
- (3 = require both keys; default)

identity: Identity string for EAP, must be in quotes.

password: Password string for EAP, must be in quotes.

ca_cert: File path to CA certificate file.

- This file can have one or more trusted CA certificates.
- If ca_cert is not included, server certificate will not be verified. This is insecure and the CA file should always be configured.

phasel: Phasel (outer authentication, i.e., TLS tunnel) parameters(string with field-value pairs, e.g., "peapver=0" or

- "peapver=1 peaplabel=1")
- 'peapver' can be used to force which PEAP version (0 or 1) is used.
- 'peaplabel=1' can be used to force new label, "client PEAP encryption",
- to be used during key derivation when PEAPv1 or newer. Most existing
- PEAPv1 implementation seem to be using the old label, "client EAP
- encryption", and wpa_supplicant is now using that as the default value.
- Some servers, e.g., Radiator, may require peaplabel=1 configuration to
- interoperate with PEAPv1; see eap_testing.txt for more details.
- 'peap_outer_success=0' can be used to terminate PEAP authentication on
- tunneled EAP-Success. This is required with some RADIUS servers that
- implement draft-josefsson-pppext-eap-tls-eap-05.txt (e.g.,
- Lucent NavisRadius v4.4.0 with PEAP in "IETF Draft 5" mode)
- include_tls_length=1 can be used to force wpa_supplicant to include
- TLS Message Length field in all TLS messages even if they are not
- fragmented.
- sim_min_num_chal=3 can be used to configure EAP-SIM to require three
- challenges (by default, it accepts 2 or 3)
- result_ind=1 can be used to enable EAP-SIM and EAP-AKA to use
- protected result indication.
- 'crypto_binding' option can be used to control PEAPv0 cryptobinding
- behavior:
 - o 0 = do not use cryptobinding (default)
 - o 1 = use cryptobinding if server supports it
 - o 2 = require cryptobinding
- EAP-WSC (WPS) uses following options: pin=<Device Password> or
- pbc=1.

phase2: (inner authentication with TLS tunnel) parameters (tring with field-value pairs, e.g., "auth=MSCHAPV2" for EAP-PEAP or

- "autheap=MSCHAPV2 autheap=MD5" for EAP-TTLS)

6.4.4 WPA (Enterprise) EAP-LEAP

The following is an example of LEAP configuration

```
# EAP-LEAP ← Comment noting this is for EAP-PEAP
network={
    ssid="NGLS"
    scan_ssid=1
    key_mgmt=WPA-EAP
    pairwise=TKIP
    group=TKIP
    auth_alg=LEAP
    eap=LEAP
    eapol_flags=0
    identity="root"
    password="password"
}
```

Network Block

ssid: SSID (mandatory); network name in one of the optional formats:

- an ASCII string with double quotation
- a hex string (two characters per octet of SSID)
- a printf-escaped ASCII string P"<escaped string>"

scan_ssid:

- 0 = do not scan this SSID with specific Probe Request frames (default)
- 1 = scan with SSID-specific Probe Request frames (this can be used to find APs that do not accept broadcast SSID or use multiple SSIDs; this will add latency to scanning, so enable this only when needed)

key_mgmt: list of accepted authenticated key management protocols

- WPA-EAP = WPA using EAP authentication
- IEEE8021X = IEEE 802.1X using EAP authentication and (optionally) dynamically generated.

pairwise: list of accepted pairwise (unicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- NONE = Use only Group Keys (deprecated, should not be included if APs support pairwise keys)
- If not set, this defaults to: CCMP TKIP

group: list of accepted group (broadcast/multicast) ciphers for WPA

- CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0, WPA2]
- TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0,WPA]
- If not set, this defaults to: CCMP TKIP WEP104 WEP40

eap: Pre-configured EAP method (TLS, TTLS, PEAP, or LEAP).

This optional field can be used to specify which EAP method will be used with this credential. If not set, the EAP method is selected automatically based on ANQP information (e.g., NAI Realm).

Continued on next Page

Continuation of network block for LEAP authentication.

eapol_flags: IEEE 802.1X/EAPOL options (bit field)

- Dynamic WEP key require for non-WPA mode
- bit0 (1): require dynamically generated unicast WEP key
- bit1 (2): require dynamically generated broadcast WEP key
- (3 = require both keys; default)

identity: Identity string for EAP, must be in quotes.

password: Password string for EAP, must be in quotes.

6.5 Activating IPv6 Transport

The Dart Vision Reader device, running appropriate firmware V5.0.0 or later, will automatically generate a link-local IPv6 address after booting up. However, in order to acquire a global IPv6 address, which is needed for IPv6 communication with the VSS server, there must be an IPv6 router on the network having the following configuration:

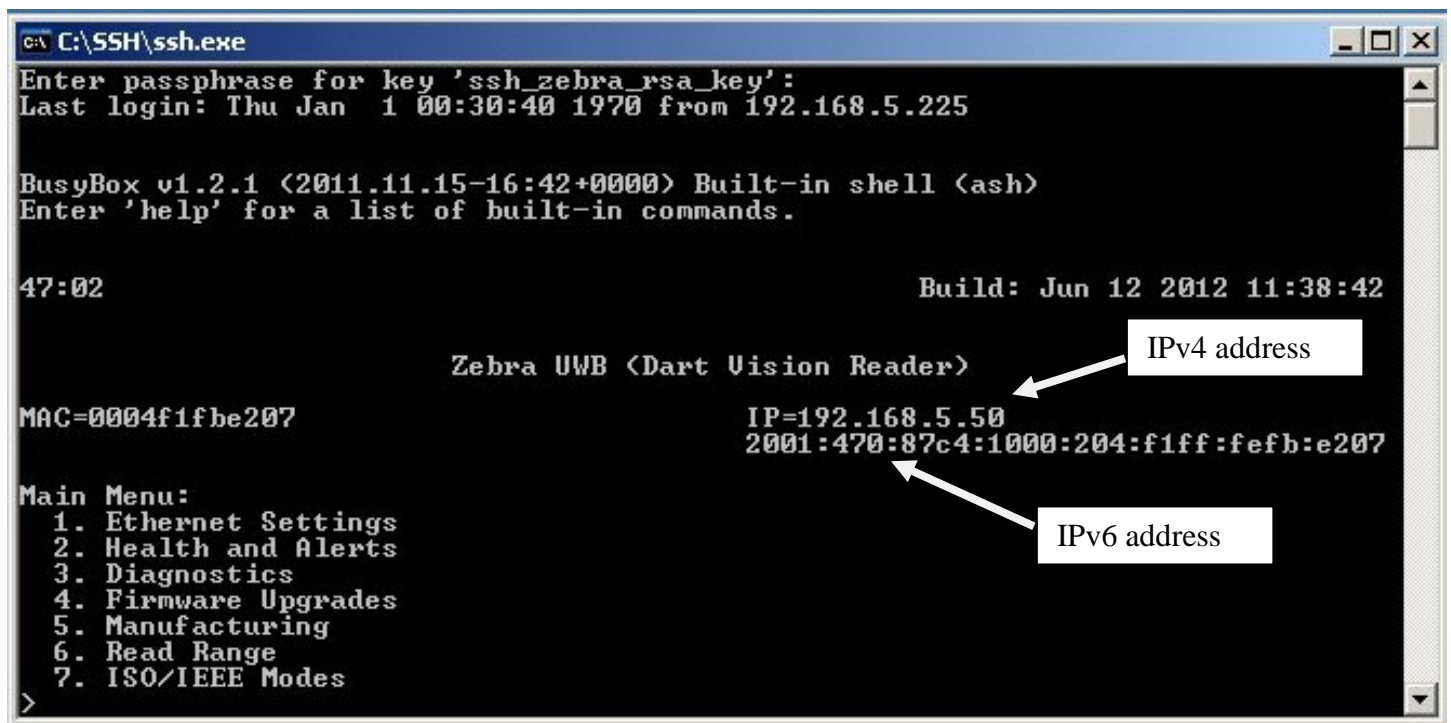
- Allow local IPv6 traffic with a global.
- IPv6 network prefix (such as, for example, 2001:470:87c4:1000::/64)
- Enable neighbor discovery/route advert. protocol

NOTE: The global IPv6 network prefix is a globally unique address and can be used inside or outside of internal networks. You need to apply for and obtain a global IPv6 network prefix for your network (for example, see www.tunnelbroker.net). This is a task typically performed by IT personnel responsible for the network.

If additionally you would like to assign a fixed IPv4 address to a Dart Vision Reader, you can do so via the sensor menu, which can be accessed via Telnet (if enabled) or SSH client, or direct connect with serial cable.²

For example, if you Telnet to a sensor, you may see three IP addresses at the top right of the window, as shown in the screenshot below:

- An IPv4 address, if one has been configured using the 'Ethernet Settings' in the sensor menu (see screenshot).
- A global IPv6 address. This is the IPv6 address required for communication with the VSS server using IPv6 transport.
- A link-local IPv6 address, automatically generated at boot up time from the MAC address of the device



```
C:\SSH\ssh.exe
Enter passphrase for key 'ssh_zebra_rsa_key':
Last login: Thu Jan  1 00:30:40 1970 from 192.168.5.225

BusyBox v1.2.1 (2011.11.15-16:42+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

47:02                                     Build: Jun 12 2012 11:38:42

                                Zebra UWB (Dart Vision Reader)
MAC=0004f1fbe207                    IP=192.168.5.50
                                      2001:470:87c4:1000:204:f1ff:febh:e207

Main Menu:
1. Ethernet Settings
2. Health and Alerts
3. Diagnostics
4. Firmware Upgrades
5. Manufacturing
6. Read Range
7. ISO/IEEE Modes
>
```

The screenshot shows an SSH terminal window titled 'C:\SSH\ssh.exe'. It displays the login process for a Zebra UWB (Dart Vision Reader) device. The terminal output includes the device's build date (Jun 12 2012 11:38:42), MAC address (0004f1fbe207), and a list of menu options. Two annotations with arrows point to the IP addresses displayed: 'IPv4 address' points to '192.168.5.50' and 'IPv6 address' points to '2001:470:87c4:1000:204:f1ff:febh:e207'.

Figure 26: IPv6 Example

6.5.1 Firmware upgrade

The unit upgrades its firmware via either a TFTP or FTP server connection through the Ethernet. The firmware is currently upgraded directly through the units' menu system, either Serial HyperTerminal or a Ethernet SSH/Telnet session. The menus and process will be the same either way.

First make sure the firmware files are located on the server TFTP or FTP root directory. The files will most likely be:

- Upgrade_dart.txt (a BASH script for handling the actual firmware files)
- uImage_dart
- u-boot.bin
- jffs2_dart.img
- initrd.boot

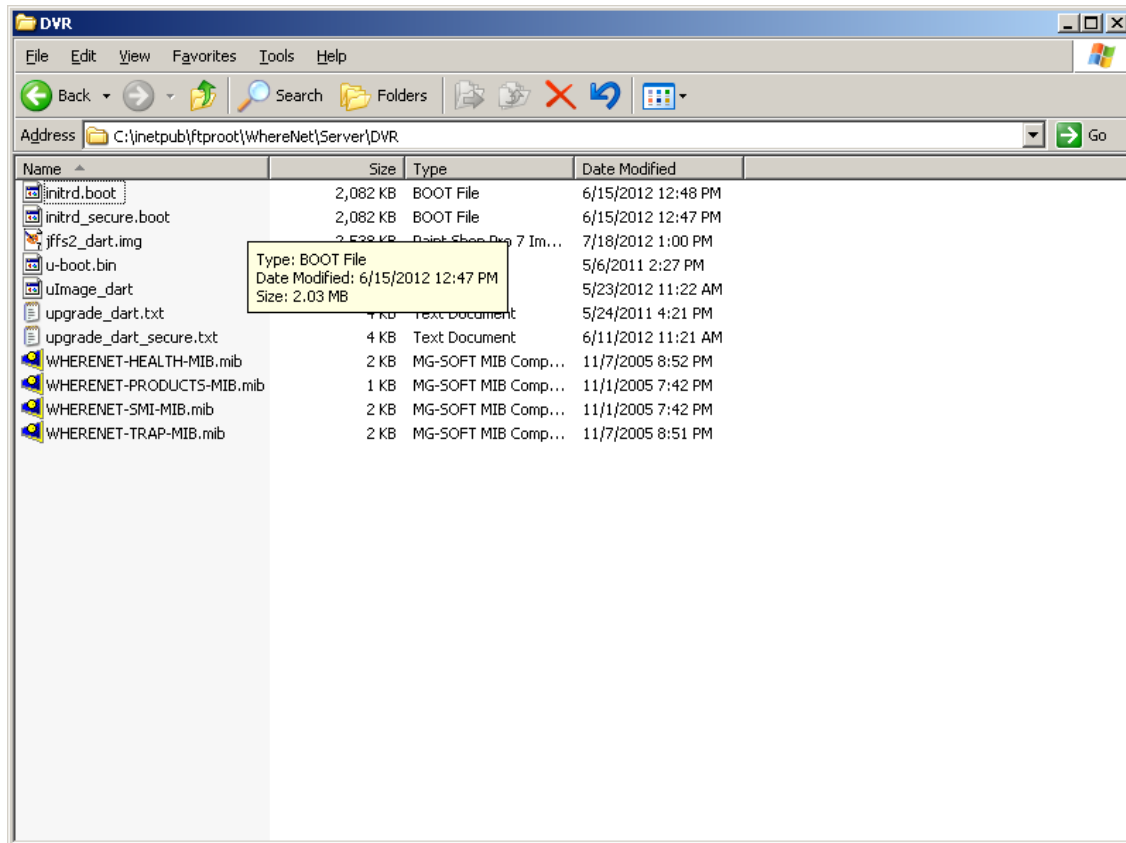
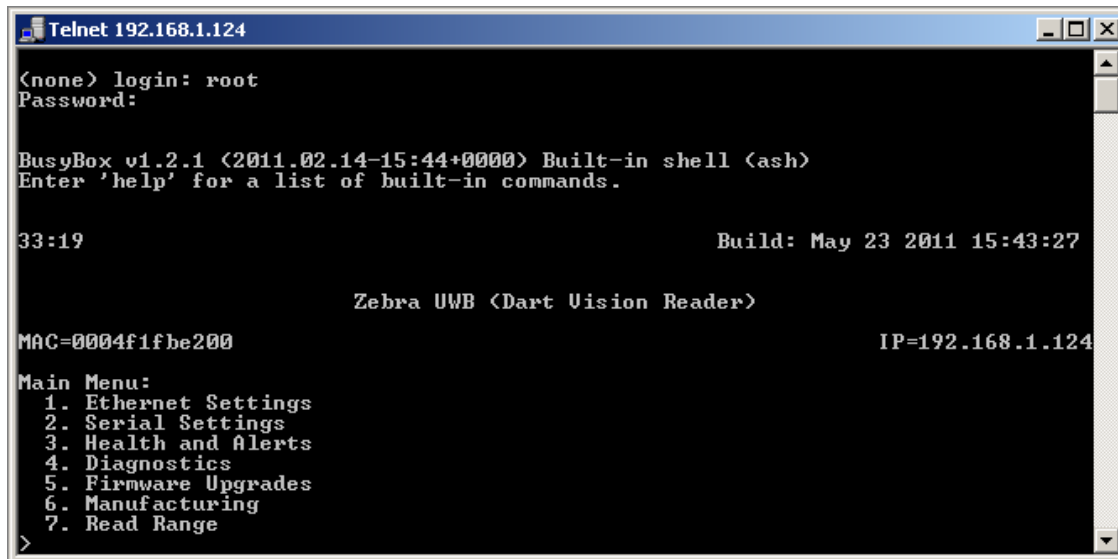


Figure 27: Firmware Upgrade Files

6.6 Firmware Upgrade Process

From the main menu select (5) Firmware Upgrades.



```

Telnet 192.168.1.124
<none> login: root
Password:

BusyBox v1.2.1 <2011.02.14-15:44+0000> Built-in shell (ash)
Enter 'help' for a list of built-in commands.

33:19                                     Build: May 23 2011 15:43:27

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe200                                IP=192.168.1.124

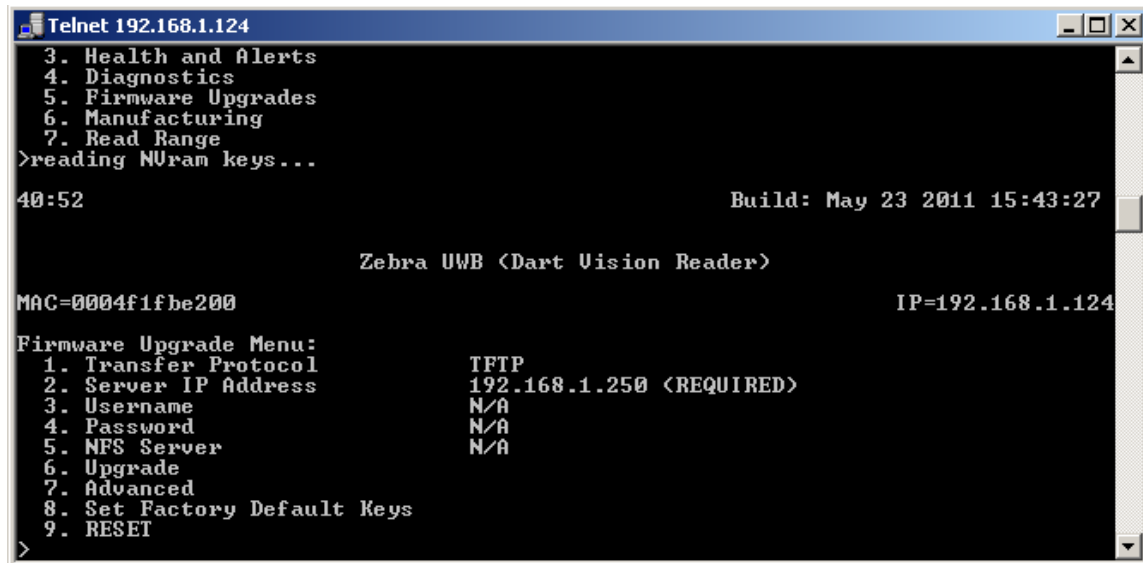
Main Menu:
 1. Ethernet Settings
 2. Serial Settings
 3. Health and Alerts
 4. Diagnostics
 5. Firmware Upgrades
 6. Manufacturing
 7. Read Range
>

```

Figure 28: Firmware Upgrade Main Menu

From the Firmware upgrade menu set either TFTP or FTP, and the Server IP address. The settings are saved by uses of the escape (Esc) key, and the password of 'ff2. *Note: After saving any changes, the user may need to re-enter the Firmware Upgrades menu from the Main Menu.*

If setting file transfer for FTP, the User and Password of the FTP server will need to be entered. For this example TFTP will be used.



```

Telnet 192.168.1.124
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>reading NVRam keys...

40:52                                     Build: May 23 2011 15:43:27

                                Zebra UWB <Dart Vision Reader>

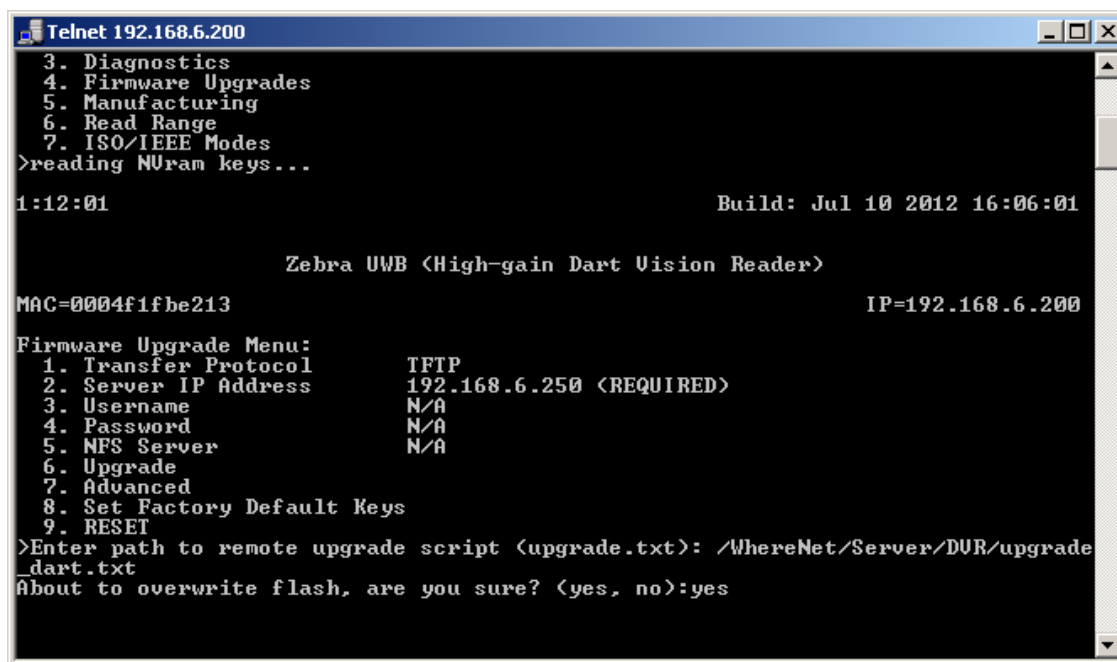
MAC=0004f1fbe200                                IP=192.168.1.124

Firmware Upgrade Menu:
 1. Transfer Protocol          TFTP
 2. Server IP Address         192.168.1.250 <REQUIRED>
 3. Username                  N/A
 4. Password                  N/A
 5. NFS Server                 N/A
 6. Upgrade
 7. Advanced
 8. Set Factory Default Keys
 9. RESET
>

```

Figure 29: Firmware Upgrade Menu

From the Firmware Upgrades Menu, select (6) Upgrade, and the unit will prompt the user for the path to the upgrade_dart.txt file. Note full path must be used if files are not directly under the ftp root directory. If all the firmware files placed in the FTP root directory, the user needs to do is press Enter/Return at this point.



```

Telnet 192.168.6.200
3. Diagnostics
4. Firmware Upgrades
5. Manufacturing
6. Read Range
7. ISO/IEEE Modes
>reading NURam keys...
1:12:01                               Build: Jul 10 2012 16:06:01

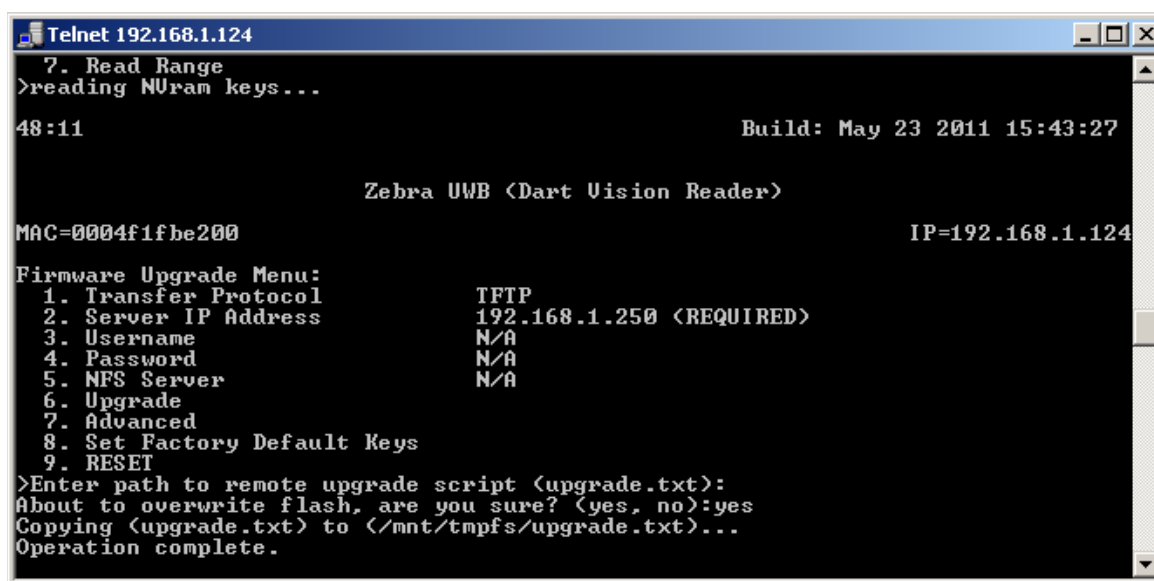
                                Zebra UWB <High-gain Dart Vision Reader>

MAC=0004f1fbe213                               IP=192.168.6.200

Firmware Upgrade Menu:
1. Transfer Protocol          TFTP
2. Server IP Address         192.168.6.250 <REQUIRED>
3. Username                  N/A
4. Password                  N/A
5. NFS Server                 N/A
6. Upgrade
7. Advanced
8. Set Factory Default Keys
9. RESET
>Enter path to remote upgrade script (upgrade.txt): /WhereNet/Server/DUR/upgrade
_dart.txt
About to overwrite flash, are you sure? <yes, no>:yes
  
```

Figure 30: Firmware Upgrade Script

The unit will prompt the user to confirm the upgrade, respond (yes) to confirm. Then the unit will respond “Operation Complete” which indicates that the unit located, download, and started the firmware upgrade process.



```

Telnet 192.168.1.124
7. Read Range
>reading NURam keys...
48:11                               Build: May 23 2011 15:43:27

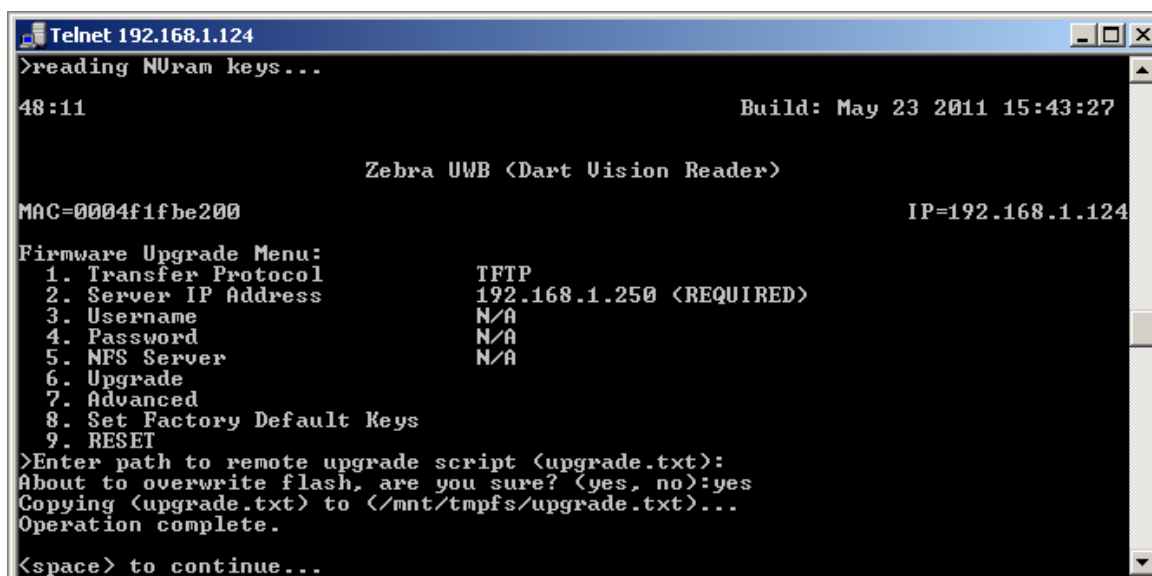
                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe200                               IP=192.168.1.124

Firmware Upgrade Menu:
1. Transfer Protocol          TFTP
2. Server IP Address         192.168.1.250 <REQUIRED>
3. Username                  N/A
4. Password                  N/A
5. NFS Server                 N/A
6. Upgrade
7. Advanced
8. Set Factory Default Keys
9. RESET
>Enter path to remote upgrade script (upgrade.txt):
About to overwrite flash, are you sure? <yes, no>:yes
Copying (upgrade.txt) to </mnt/tmpfs/upgrade.txt>...
Operation complete.
  
```

Figure 31: Firmware Upgrade Response

When the unit has completed the firmware upgrade process it will respond with “<space> to continue...” After that the unit requires a RESET (9) for the new firmware to take effect.



```

Telnet 192.168.1.124
>reading NURam keys...
48:11                               Build: May 23 2011 15:43:27

                                Zebra UWB (Dart Vision Reader)

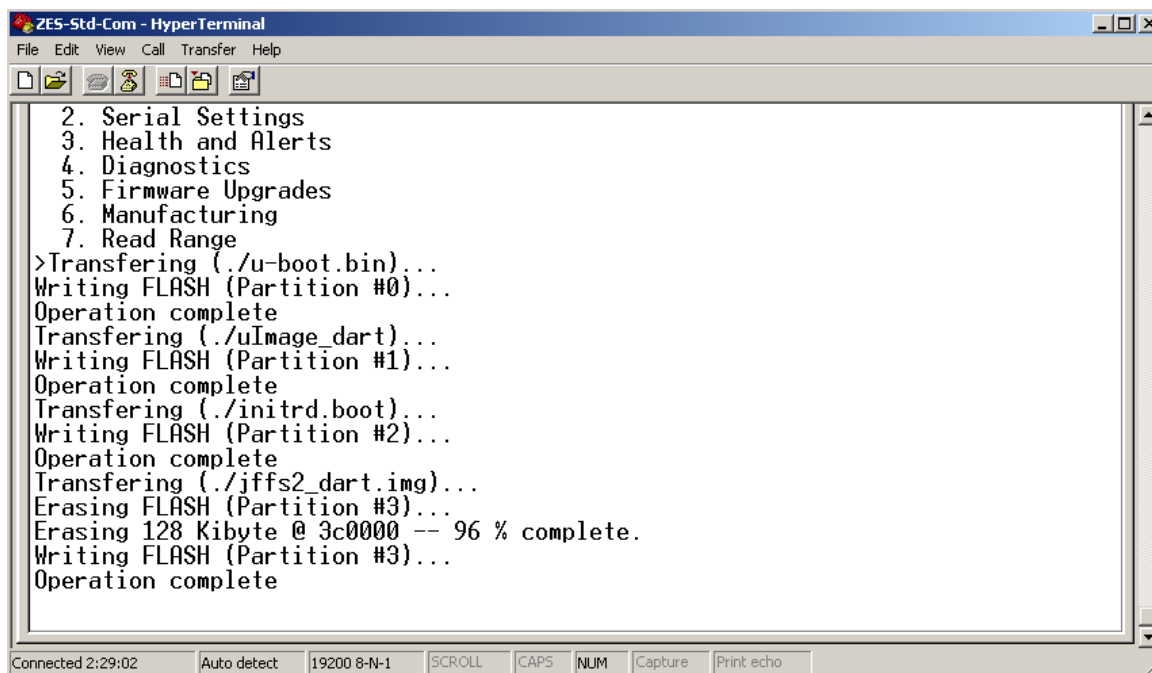
MAC=0004f1f8e200                               IP=192.168.1.124

Firmware Upgrade Menu:
1. Transfer Protocol           TFTP
2. Server IP Address          192.168.1.250 (REQUIRED)
3. Username                   N/A
4. Password                   N/A
5. NFS Server                 N/A
6. Upgrade
7. Advanced
8. Set Factory Default Keys
9. RESET
>Enter path to remote upgrade script (upgrade.txt):
About to overwrite flash, are you sure? (yes, no):yes
Copying (upgrade.txt) to (/mnt/tmpfs/upgrade.txt)...
Operation complete.
<space> to continue...

```

Figure 32: Firmware Upgrade Complete

If doing the operation via a serial port and HyperTerminal connection, the user will see the firmware files being individually downloaded and written by the unit. However in this method once the unit responds “Operation Complete” the user will need to press the Escape (Esc) key, and then RESET the unit.



```

ZES-Std-Com - HyperTerminal
File Edit View Call Transfer Help

2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>Transferring (./u-boot.bin)...
Writing FLASH (Partition #0)...
Operation complete
Transferring (./uImage_dart)...
Writing FLASH (Partition #1)...
Operation complete
Transferring (./initrd.boot)...
Writing FLASH (Partition #2)...
Operation complete
Transferring (./jffs2_dart.img)...
Erasing FLASH (Partition #3)...
Erasing 128 Kibyte @ 3c0000 -- 96 % complete.
Writing FLASH (Partition #3)...
Operation complete

Connected 2:29:02   Auto detect   19200 8-N-1   SCROLL   CAPS   NUM   Capture   Print echo

```

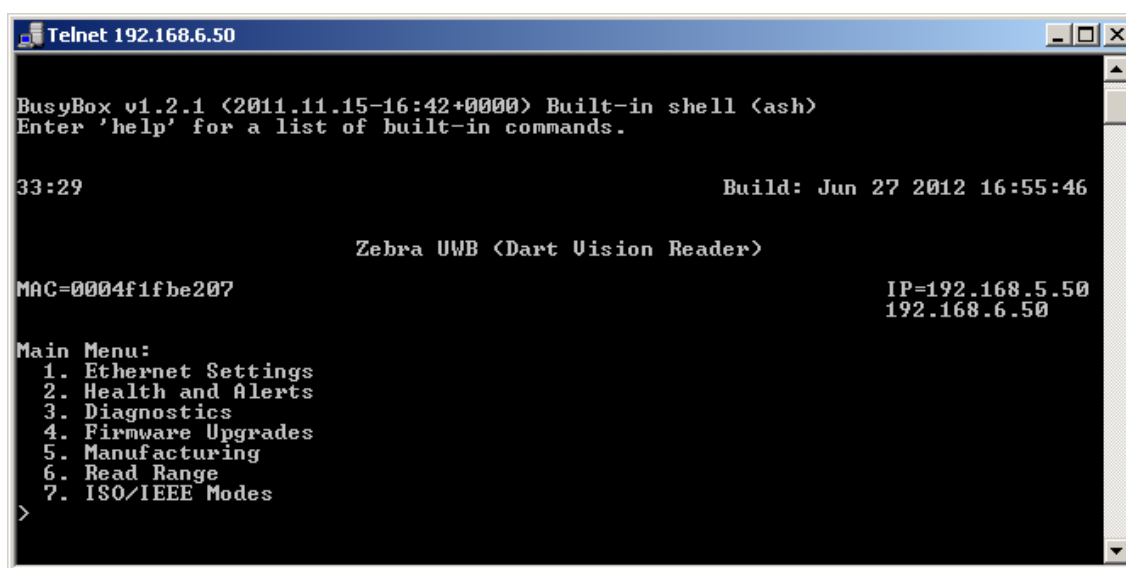
Figure 33: Firmware Upgrade Serial

6.7 Read Range

The Read Range menu, accessed from the Main Menu via Read Range item selection, allows the user to set the tag Read Range (or reception range) in a quantized read/reception range. In firmware prior to Build Date (July 10 2012) only a coarse Reader Range from 1 to 25 is available and the default is set to the maximum range of 25.

In the latest version of Firmware V5.0.0, Build Date (July 10 2012) and later, Reader Range will have two modes available, the Coarse Range from 1 to 25, and a Fine Mode from 1 to 54.

The following steps will be valid through either Telnet or Hyper Terminal. The Range setting is real time and does not require a RESET.

A screenshot of a Telnet window titled 'Telnet 192.168.6.50'. The window shows a BusyBox v1.2.1 shell prompt. The output displays system information: '33:29' and 'Build: Jun 27 2012 16:55:46'. It identifies the device as 'Zebra UWB <Dart Vision Reader>' with MAC address 'MAC=0004f1fbe207' and IP address 'IP=192.168.5.50' and '192.168.6.50'. A 'Main Menu:' is listed with seven options: 1. Ethernet Settings, 2. Health and Alerts, 3. Diagnostics, 4. Firmware Upgrades, 5. Manufacturing, 6. Read Range, and 7. ISO/IEEE Modes. A greater-than sign (>) is shown at the bottom of the menu list.

```
Telnet 192.168.6.50

BusyBox v1.2.1 (2011.11.15-16:42+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

33:29                                     Build: Jun 27 2012 16:55:46

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                        IP=192.168.5.50
                                         192.168.6.50

Main Menu:
 1. Ethernet Settings
 2. Health and Alerts
 3. Diagnostics
 4. Firmware Upgrades
 5. Manufacturing
 6. Read Range
 7. ISO/IEEE Modes
>
```

Figure 34: Read Range Main Menu

Setting the Range Value can be accomplished in a couple of ways:

- The first way is to directly enter a value under (1) Range.
- The second is to use the (+/-) keys to increment or decrement the Range Value.

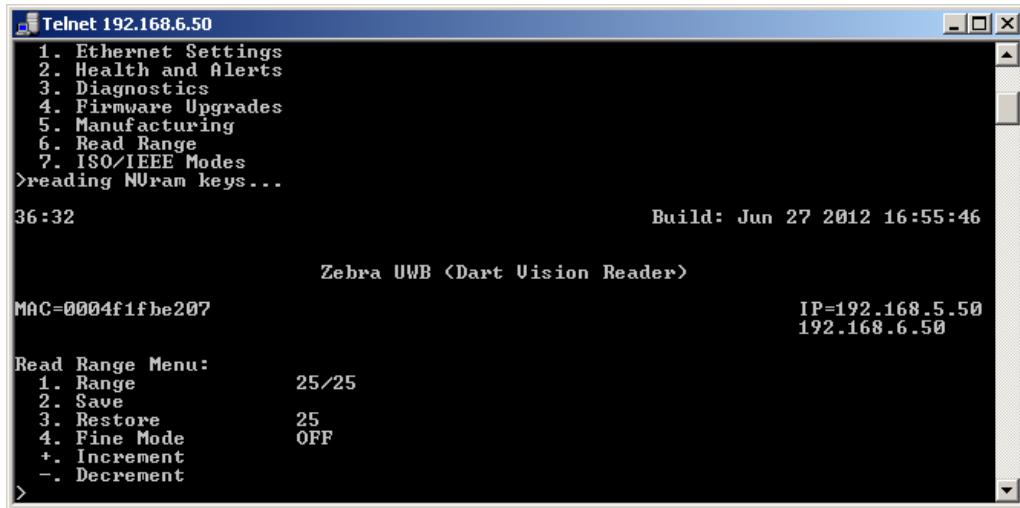


Figure 35: Read Range Menu

Direct Range Entry, Range (1), and enter a value from 1 to 25.

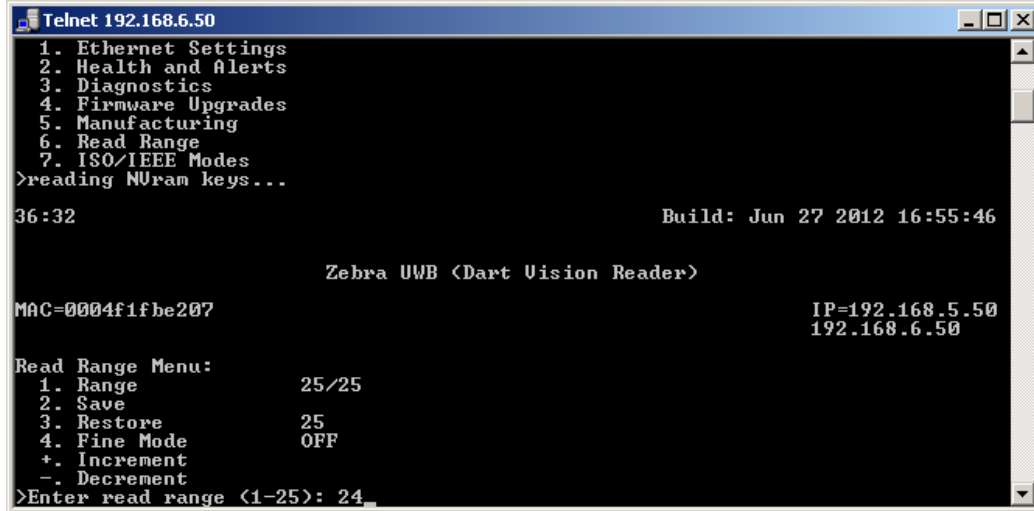
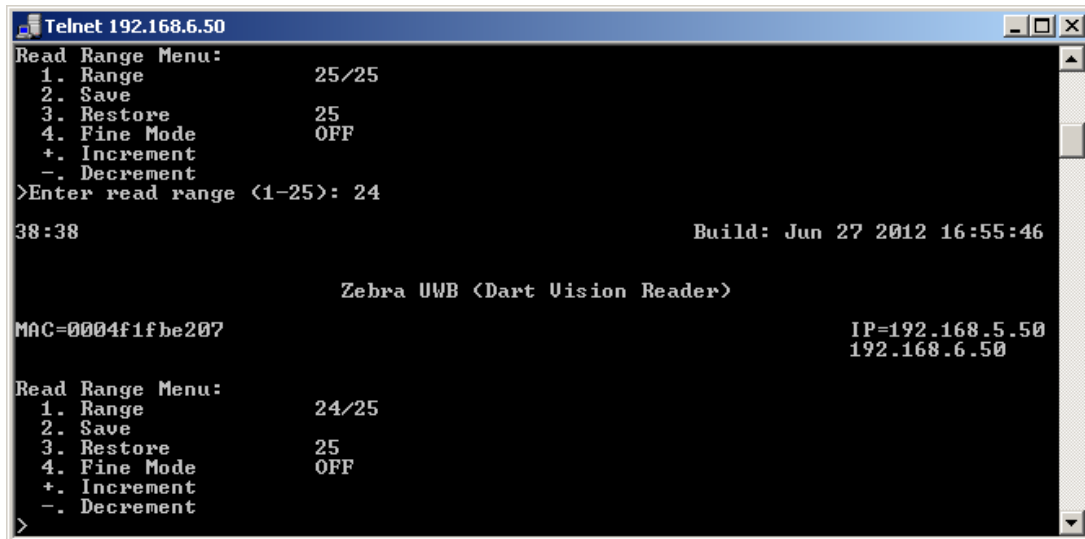


Figure 36: Read Range Direct Entry

After pressing Enter/Return, the screen will display the Range value set.



```

Telnet 192.168.6.50
Read Range Menu:
1. Range          25/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>Enter read range <1-25>: 24
38:38                               Build: Jun 27 2012 16:55:46

                                Zebra UWB <Dart Vision Reader>

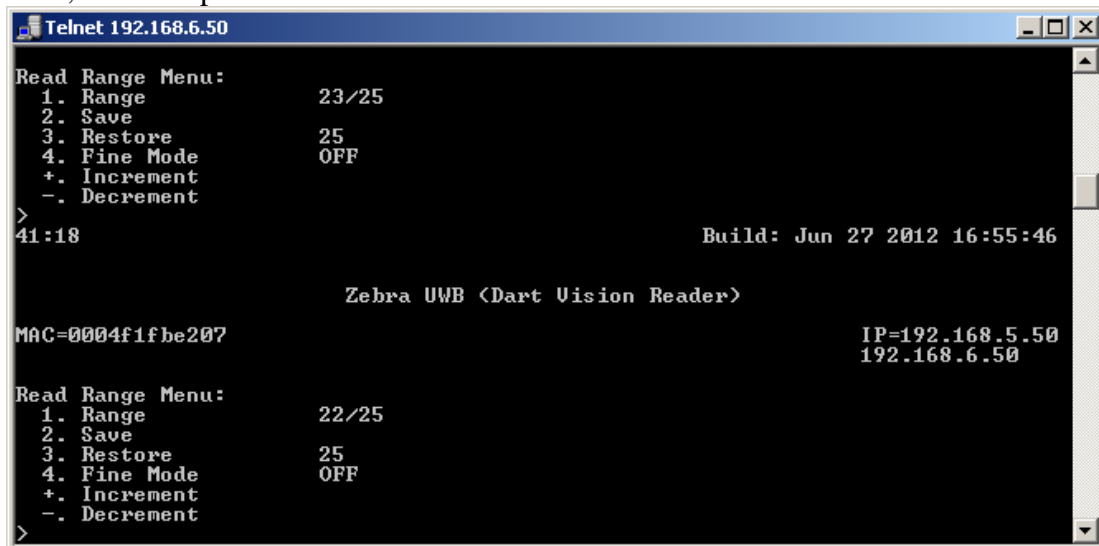
MAC=0004f1fbe207                    IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range          24/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>

```

Figure 37: Read Range Value

Using the (+ or -) key from this point the value will increment or decrement by 1 for each press of the key. In example below pressed the (-) key two times and the value decreased to 22, from the previous value of 24.



```

Telnet 192.168.6.50
Read Range Menu:
1. Range          23/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>
41:18                               Build: Jun 27 2012 16:55:46

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                    IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range          22/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>

```

Figure 38: Read Range Increment/Decrement

Once a Range value has been established, then the user must save the value for the setting to stay after a reboot of the unit.

Using (2) Save, and confirming with 'yes' the value will be saved into memory. The user will be prompted to "<space> to continue...."

```

Telnet 192.168.6.50
Read Range Menu:
1. Range          23/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>
41:18                               Build: Jun 27 2012 16:55:46

                                Zebra UWB (Dart Vision Reader)

MAC=0004f1fbe207                    IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range          22/25
2. Save
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>Save read range <yes,no>? yes_

```

Figure 39: Read Range Save

Additional Read Range Menu Items:

Restore: is used to restore the previously saved value, if the user has altered the value sense the last time that it was saved.

Fine Mode: is used to switch the Read Range control to have greater granularity than in coarse mode. Just like the coarse mode , the value can be entered directly and/or the value can be incremented or decremented in steps of 1 by the (+/-) keys. Save the set range in Fine Mode is also the same. Enable by selecting item number that corresponds to Fine Mode and enter 'yes' to enable .

```

Telnet 192.168.6.50
3. Restore        25
4. Fine Mode      OFF
+. Increment
-. Decrement
>Save read range <yes,no>? yes
Done
<space> to continue...

1:29:47                               Build: Jun 27 2012 16:55:46

                                Zebra UWB (Dart Vision Reader)

MAC=0004f1fbe207                    IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range          22/25
2. Save
3. Restore        22
4. Fine Mode      OFF
+. Increment
-. Decrement
>
Fine range steps <yes,no>? yes_

```

Figure 40: Fine Read Range

The unit will now display the fine mode range. The (~) before the Fine Mode Range number indicates that it was switched to fine mode from normal. The (~) will disappear if the fine number is directly entered or is selected with the (+/-) keys.

```

Telnet 192.168.6.50
4. Fine Mode          OFF
+. Increment
-. Decrement
>
Fine range steps <yes,no>? yes
1:31:03                               Build: Jun 27 2012 16:55:46

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                      IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range              ~42/54
2. Save
3. Restore            ~42
4. Fine Mode          ON
+. Increment
-. Decrement

** You have transitioned to fine mode.
   You will be able to save this upon ESC **
>

```

Figure 41: Read Range Fine Set

For the unit to remain in this mode after a Reset or Reboot this setting must be saved. Escape key (ESC), and enter 'yes' to save.

```

Telnet 192.168.6.50
4. Fine Mode          OFF
+. Increment
-. Decrement
>
Fine range steps <yes,no>? yes
1:31:03                               Build: Jun 27 2012 16:55:46

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe207                      IP=192.168.5.50
                                      192.168.6.50

Read Range Menu:
1. Range              ~42/54
2. Save
3. Restore            ~42
4. Fine Mode          ON
+. Increment
-. Decrement

** You have transitioned to fine mode.
   You will be able to save this upon ESC **
>Save fine mode setting <yes,no>? yes

```

Figure 42: Read Range Fine Save

6.7.1 Estimated Read Range Per Range Setting

The following figures show the estimated Dart Vision Reader UWB range based on the UWB Range setting.

Course Read Range Settings Graph

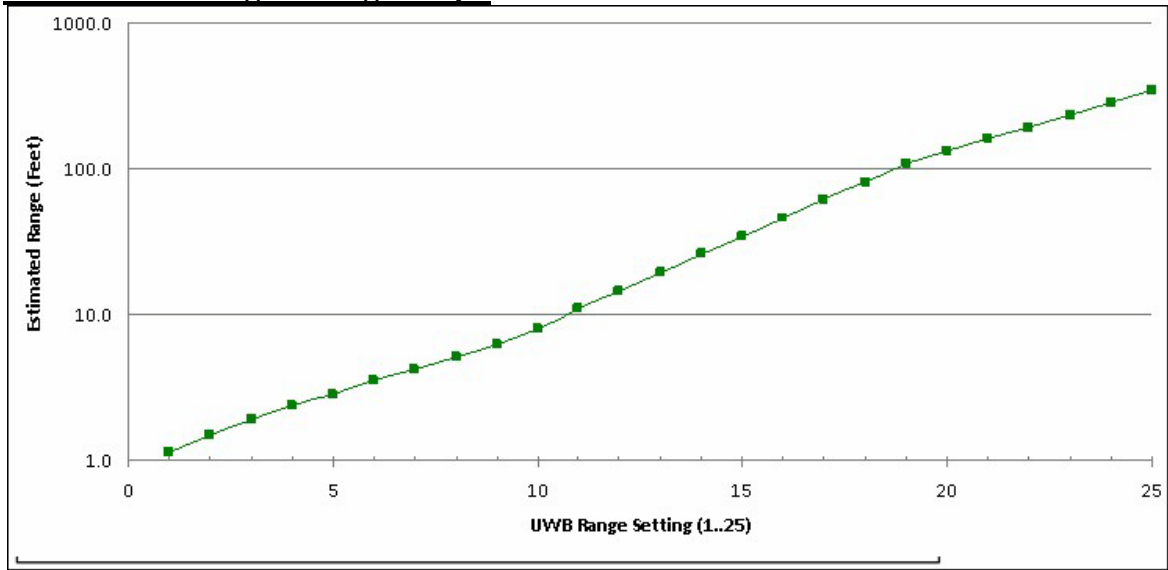


Figure 43: Read Range Course Distance

Fine Read Range Settings Graph

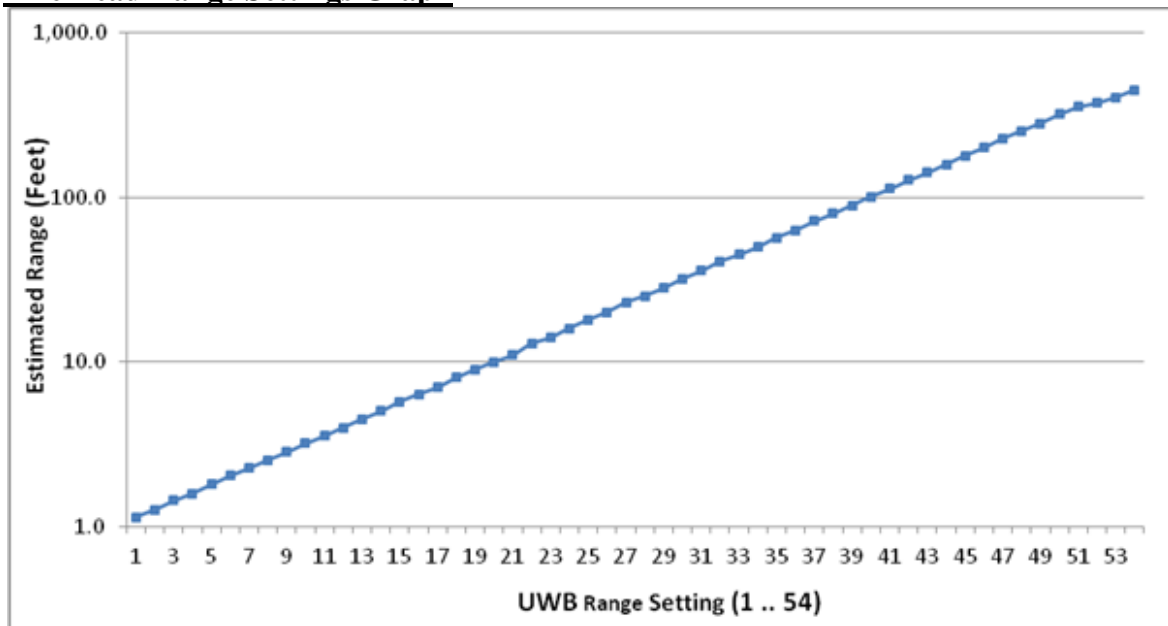


Figure 44: Read Range Fine Distance

7. OUTPUT DATA DESCRIPTION

A Dart Vision Reader with firmware prior to Build Date (July 10 2012) will require a client program using stream communication protocol is required to retrieve data from the reader. Any client program connecting to the reader must use port id 5117. All data and status messages will be sent over the LAN interface as ASCII strings.

DVR software V5.0.0 Build Data (July 10 2012) or later will support three output types:

- DVR output data, backward compatible with current DVR output
- Zebra SLMP with Zebra additional SLMP format. “Z-SLMF” (SLMF, text based, TCP/IP ISO24730-1, with provision for reporting non-ISO, non-IEEE tag IDs from Dart mode tags).

7.1 Dart VISION Reader Output (Port 5117)

When a connection is established on port 5117, the Dart Vision Reader sends the following messages types:

Tag packets

Tag detection output format for a tag packet is as follows:

P, tag_ID, battery <LF>

Where each of these fields are described as follows:

p	Start of packet message header.
<tag_ID>	Tag ID, represented in ASCII, either 8, 12 or 16 characters.
<battery>	Indicates the battery level of the tag (00 - 15) where 15 is maximum battery level.

d-Packets

d-packets are sent to the output stream every minute to indicate reader status. These messages are useful for continuous monitoring of the Dart Vision Reader. The message output format for a diagnostic packet (d-packet) is as follows:

d, sequence_count, pkt_ID <LF>

Where each of these fields are described as follows:

D	d-packet message header indicating the start of a new diagnostic packet.; these packets are sent every 60 seconds by default.
<sequence_count>	Wrap-around counter, ranging from 00 to FF, value increments by one for each d-packet sent; this d-packet is used to inform the client that the tag reader is alive even when no tag data is within detection range.
<pkt_ID>	Message header ID. (Set to 01 for this application).

7.2 Dart VISION Reader Read Range Port (Port 5110)

In version 5.0.1 Build Date Aug 15 2012 or later provides Real Time adjustment of Read Range setting.

- The real time setting of Read Range is accomplished via a telnet connection through port 5110, and takes effect in real time. (Example from a command prompt: **telnet 192.168.6.200 5110**)
- These adjustments are in the Fine Reader Range Steps.
- Read Range commands are as follows. (All in lower case)
 - To display current setting 'range'
 - To increase or decrease Read Range in increments of 1, is 'range +' or 'range -'
 - To set a desired range directly it is range and then a number, for example 'range 46'
 - To save a set range, 'range save'
- The unit will respond with either an "ACK" accepted or a "NACK" not accepted for, or the numeric setting of the Read Range
- commands.
- Note if no commands are entered telnet session times out and closes after 1 minute.
-

Following is example of the telnet session range entries.

```

Telnet 192.168.6.200
nack
  range
36
  range +
ack
  range
37
  range 46
ack
  range
46
  range -
ack
  range
45
  range sanve
nack
  range save
ack
  -
  
```

Figure 45: Port 5110 Commands

7.3 Dart VISION Reader Z-SLMP Output (Port 5118)

Message Header

When a connection is established on port 5118, the Dart Vision Reader sends the following messages on the new connection:

```
DVR,Z-SLMF,Drft120319,1.0>Welcome to the DVR Enhanced Text Stream Interface.
FieldDefinition,Source,String
FieldDefinition,Format,String
FieldDefinition,Tag_ID_Format,HexBinary
FieldDefinition,Tag_ID,HexBinary
FieldDefinition,X,Double
FieldDefinition,Y,Double
FieldDefinition,Z,Double
FieldDefinition,Battery,HexBinary
FieldDefinition,Timestamp,DateTime
FieldDefinition,DVR_Name,String
FieldDefinition,Temperature,String
FieldDefinition,Period,Integer
LocateMessageDefinition,DVR,DFT,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp
LocateMessageDefinition,DVR,DFT+T,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp,Temperature
LocateMessageDefinition,DVR,DFT+T,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp,DVR_Name
LocateMessageDefinition,DVR,DFT+DT,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp,DVR_Name,Temperature
KeepAlive,60
```

Subsequent DVR messages are detected tag presence and keep alive diagnostic messages on the connection using the default format (DFT) as defined in ISO standard (24730-1)

Tag Presence Message

DVR tag presence message format:

```
<Source>,<Format>,<Tag_ID_Format>,<Tag_ID>,<X>,<Y>,<Z>,<Battery>,<Timestamp><CR><LF>
DVR,DFT+[DT],Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp,<DVR_Name>1,<Temperature>1<CR><LF>
```

¹Optional Field where

[D] indicates <DVR_Name> included

[T] indicates <Temperature> included

DVR Example Z-SLMP (Port 5118) data:

For a Dart ISO tag:

```
DVR,DFT,0003AABBCCDD,01,,,64,2010-11-24T09:07:04-08:00
DVR,DFT+T,0003AABBCCDD,01,,,64,2010-11-24T09:07:04-08:00,22
DVR,DFT+DT,0003AABBCCDD,01,,,64,2010-11-24T09:07:04-08:00,DVR_name,22
```

For a Dart IEEE tag:

```
DVR,DFT,7493A403AABBCCDD,03,,,64,2010-11-24T09:07:04-08:00
DVR,DFT+T,7493A403AABBCCDD,03,,,64,2010-11-24T09:07:04-08:00,22
DVR,DFT+DT,7493A403AABBCCDD,03,,,64,2010-11-24T09:07:04-08:00,DVR_name,22
```

Dart tag: "FF" to indicate TagID format

```
DVR,DFT,AABBCCDD,FF,,,64,2010-11-24T09:07:04-08:00
DVR,DFT+T,AABBCCDD,FF,,,64,2010-11-24T09:07:04-08:00,22
DVR,DFT+DT,AABBCCDD,FF,,,64,2010-11-24T09:07:04-08:00,DVR_name,22
```

7.4 Dart VISION Reader SLMP Output (Port 5119)

Message Header

Once a connection is established on port 5119, the DartWand sends the following message header on the new connection:

```
DVR,SLMF,Drft120319,1.0>Welcome to the DVR Text Stream Interface.
FieldDefinition,Source,String
FieldDefinition,Format,String
FieldDefinition,Tag_ID_Format,HexBinary
FieldDefinition,Tag_ID,HexBinary
FieldDefinition,X,Double
FieldDefinition,Y,Double
FieldDefinition,Z,Double
FieldDefinition,Battery,HexBinary
FieldDefinition,Timestamp,DateTime
FieldDefinition,Period,Integer
LocateMessageDefinition,DVR,DFT,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp
KeepAlive,60
```

Subsequent DVR messages are detected tag presence and keep alive diagnostic messages on the connection using the default format (DFT) as defined in ISO standard (24730-1). This port supports ISO/IEC 15963 and IEEE EUI-64 ID formats. Legacy Dart IDs are not supported.

Tag Presence Message

The DVR sends tag presence messages with the following format:

```
<Source>,<Format>,<Tag_ID_Format>,<Tag_ID>,<X>,<Y>,<Z>,<Battery>,<Timestamp><CR><LF>
DVR,DFT,Tag_ID_Format,Tag_ID,X,Y,Z,Battery,Timestamp
```

DVR Example SLMP (Port 5119) data:

For a Dart ISO tag:

```
DVR,DFT,0003AABBCCDD,01,,,,64,2010-11-24T09:07:04-08:00,<CR><LF>
```

For a Dart IEEE tag:

```
DVR,DFT,7493A403AABBCCDD,03,,,,64,2010-11-24T09:07:04-08:00,<CR><LF>
```

8. REGULATORY COMPLIANCE INFORMATION

RF Notice

Any changes or modifications to Zebra Technologies (Zebra) equipment not expressly approved by Zebra could void the user's authority to operate the equipment.

FCC Compliance Statement

This device complies with Part 15 rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference; (2) this device must accept any interference which may cause undesired operation.

This equipment has been tested and found to comply with the limits for Class A devices, pursuant to Part 15 of the FCC Rules & Regulations.

Canadian DOC Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

9. REGULATORY COMPLIANCE INFORMATION

WLM54AG MODULE

RF Notice

Any changes or modifications to Zebra Technologies (Zebra) equipment not expressly approved by Zebra could void the user's authority to operate the equipment.

FCC Compliance Statement

This device complies with Part 15 rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference; (2) this device must accept any interference which may cause undesired operation.

FCC ID: XWX-05WLM54AG

This equipment has been tested and found to comply with the limits for Class A devices, pursuant to Part 15 of the FCC Rules & Regulations.

Canadian DOC Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

IC: 8701A-05WLM54AG

RF Exposure Statement


This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator and your body. This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

EU Compliance Information

Approved for use in the following countries.

AT	BE	BG	CY	CZ	DK	EE
FI	FR	DE	GR	HU	IE	IT
LV	LT	LU	MT	NL	PL	PT
RO	SK	SI	ES	SE	GB	
IS	LI	NO	CH		TR	

Note: See below for limitations.

	<p>Important Notice: This RF device is intended for indoor and outdoor use in all EU and EFTA with the following limitations.</p> <p>France: Outdoor use limited to 10 mW e.i.r.p. within the band 2454-2483.5 MHz</p> <p>Italy: For private use, a general authorization is required if WAS/RLAN's are used outside own premises. For public use, a general authorization is required.</p> <p>Luxembourg: General authorization required for network and service supply.</p> <p>Norway: Wideband Data Transmission systems 2400.0-2483.5 MHz does not apply for the geographical area within a radius of 20 km from the centre of Ny-Ålesund</p>
---	--

Antenna Options

This mini PCI card has been tested and found to be in compliance when used with the following 2.4 GHz antennas:

- Cisco Aironet 13.5-dBi Yagi Antenna (P/N AIR-ANT1949)
Note: Output power must be set to ≤ 6.5 dBm when using the Yagi antenna. Operating at a higher power violates national regulations. See Important Notice (above) for additional restrictions.
- Cisco Aironet 5.2-dBi Dipole Omni-directional Antenna (P/N AIR-ANT2506)
- Cisco Aironet 2-dBi Dipole Omni-directional Antenna (P/N AIR-ANT4941)

APPENDIX A

A.1 Mounting Instructions

A.1.1 Installation to Metal Strut.

The Dart Reader can be mounted to metal strut by means of the mounting bracket attached to all Readers using customer supplied hardware. This hardware could be the standard strut t-nut and corresponding bolt. Examples of this setup are shown in Figure 5 and Figure 6.

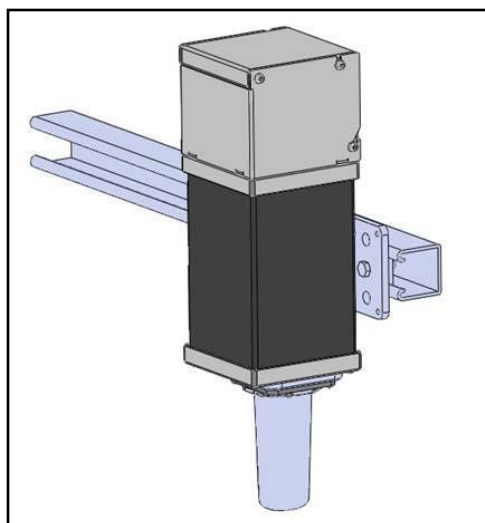


Figure 46: Installation to horizontal metal strut

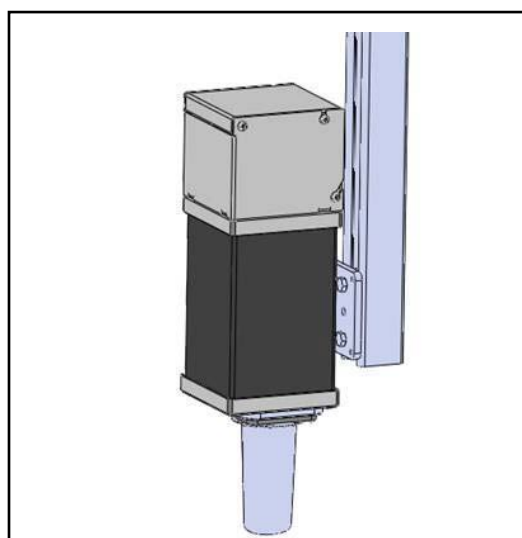


Figure 47: Vertical mounting to metal strut

A.1.2 Installation using optional mounting bracket (UM-120-00)

The optional mounting bracket shown in Figure 7 below attaches to the Reader by first removing the two nuts supplied on the Reader's bracket, see Figure 8, and then inserting the two threaded studs of the plate into the corresponding holes in the optional mounting bracket as shown in Figure 9. Using a 7/16 wrench, re-install nuts and tighten snugly.



Figure 48: Optional Mounting Bracket (UM-120-00)

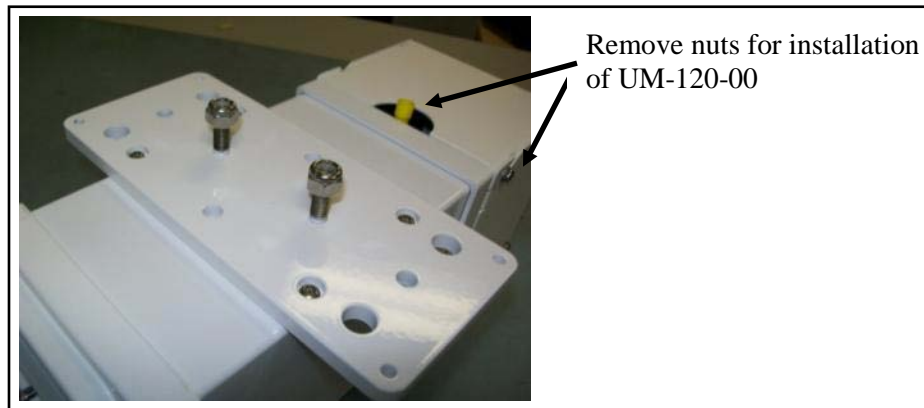


Figure 49: Removal of two nuts prior to installation

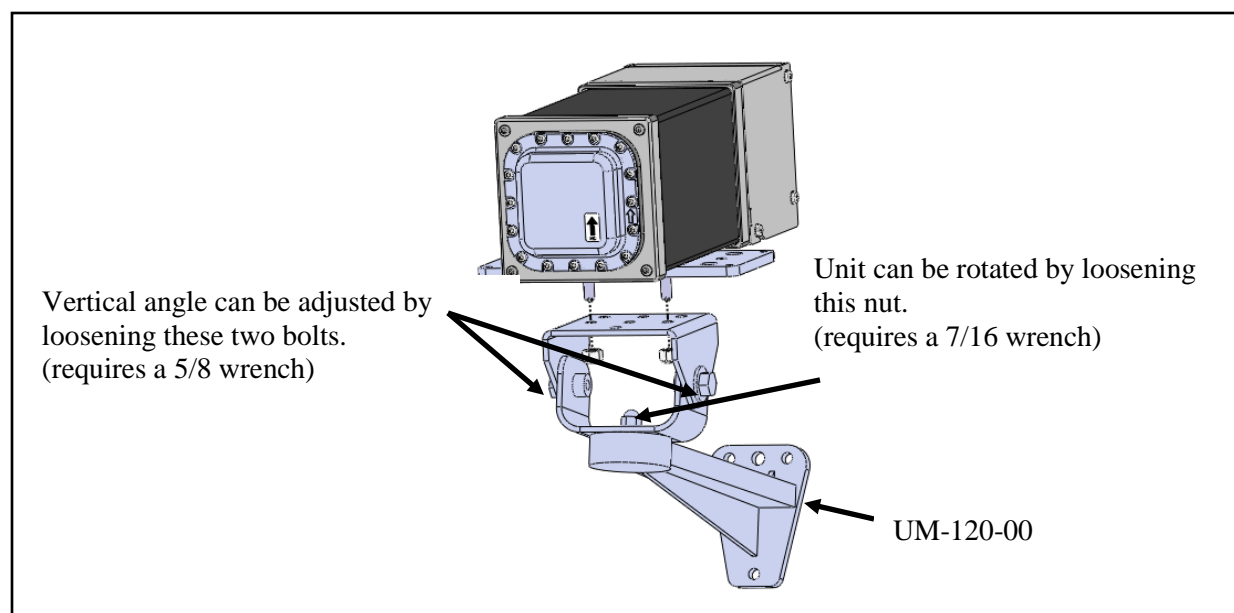


Figure 50: Attachment to optional mounting bracket



Figure 51: Mid/ High-Gain with optional mounting bracket



Figure 52: Omni with optional mounting bracket

A.1.3 Safety lanyard mounting.

A lanyard can be secured to any of the four holes not being used for mounting purposes. A typical example is shown below in Figure 12.



Figure 53: Safety lanyard attachment

APPENDIX B

B.1 Connector Removal

Depending on system configuration (refer to section 4.3 of this document) remove one or two of the connectors by turning counter clockwise. One connector is removed as shown below in Figure 13. Once removed, loosen dome nut on connector and remove rubber plug shown in Figure 14.



Figure 54: Removal of connector from unit prior to installing cable

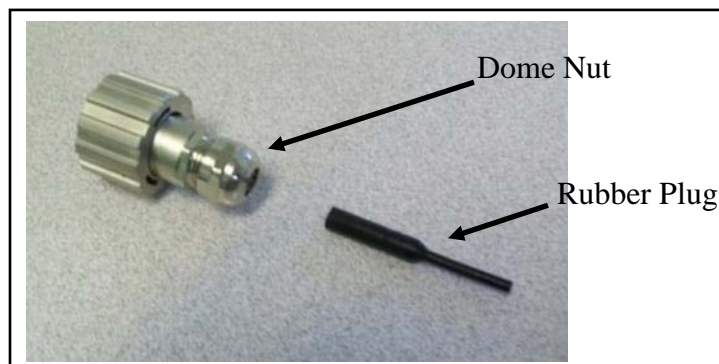


Figure 55: Removing rubber plug

B.2

Cable Preparation

B.2.1 Once the rubber plug is removed from the connector, the Ethernet cable can be inserted into the connector as shown in Figure 15. Pull cable through any distance required for comfortable access of installing the RJ45 connector to the cable as this will be pushed back through after connector is installed.



Figure 56: Insertion of Ethernet cable into connector

B.2.2 Cut back the cable jacket as shown below in Figure 16. After the jacket is cut back, the conductor pairs should be untwisted and aligned side-by-side and trimmed as shown in Figure 17 according to EIA/TIA T568B shown in Figure 18.

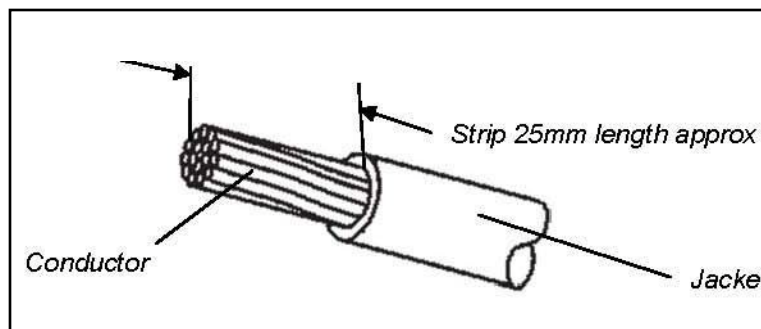


Figure 57: Cutting back of cable jacket

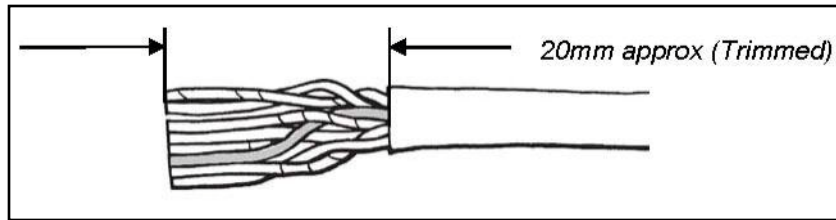


Figure 58: Alignment and trimming of wires

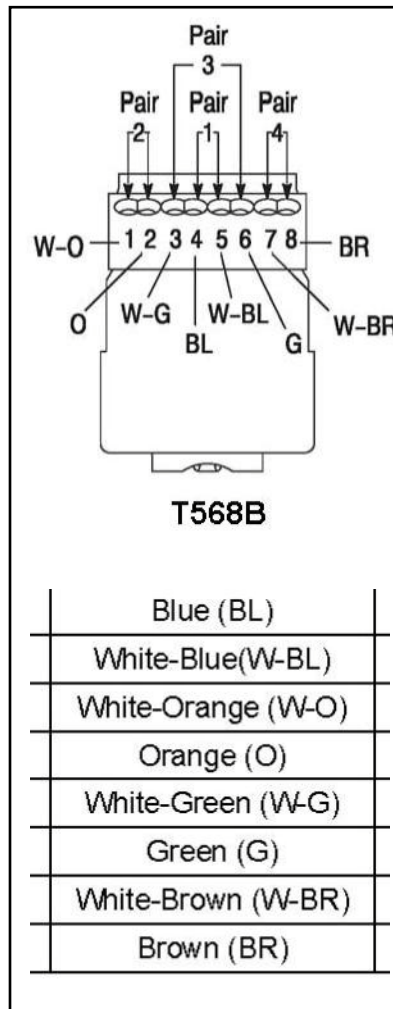


Figure 59: EIA/TIA T568B

B.2.3 After inserting the wires into the appropriate positions of the load bar, slide the cable to a point where the jacket hits against the notch of the load bar as shown in Figure 19. Trim the remaining wire ends to approximately 5mm length of the wire tips as shown in Detail A of Figure 19. Retract the cable, leaving about 1mm length of the wire tips as shown in Detail B of Figure 19.

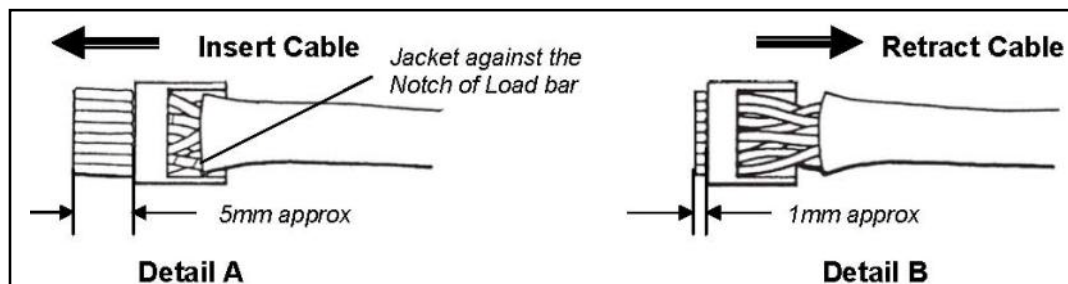


Figure 60: Load bar details

B.2.4 Insert the wired load bar into the RJ45 plug all the way until the wire tips are seated against the inside wall of the plug housing see Figure 20.

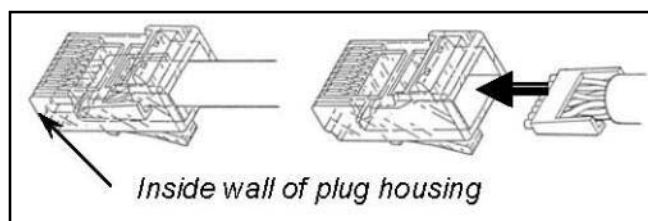


Figure 61: Inserting load bar into plug

B.2.5 Terminate the cable and the RJ45 plug with the appropriate modular plug termination tool as shown in Figure 21. Depress the locking tab of the plug, insert the plug and cable into the termination head up to the end of the inside plug housing wall and terminate. Depress the locking tab of the plug from the tool after termination.

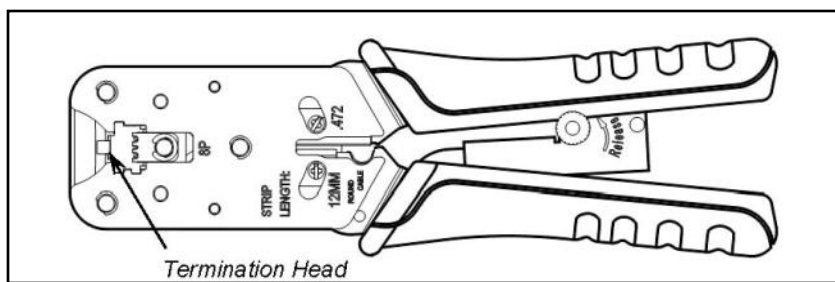


Figure 62: Modular Plug termination tool.

B.2.6 Assemble the RJ45 Plug Housing by depressing the locking tab of the RJ45 plug and align it with the wide slot in the plug housing shown in Detail A below in Figure 22. Gently pull the cable until the plug is fully seated. Hold the plug in position and rotate the domed cable fitting until tightened to a torque of 20 in-lbs [2.27 Nm] as shown in Detail B of Figure 22.

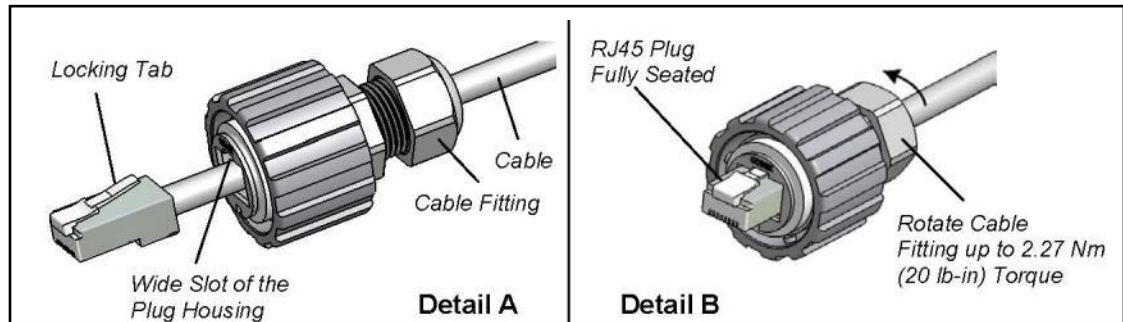


Figure 63: Assembly of RJ45 Plug Housing

B.2.7 Jack and Plug Engagement. Gently insert the assembled plug (Bayonet) into the Jack adaptor of the RJ45 receptacle, align the 3 keys of the bayonet coupling ring with the 3 bayonet channels of the receptacle and rotate the bayonet of the coupling ring until the 3 keys "click" into the bayonet channels. See Figure 23. Use of spanner wrench (see tools required section) can greatly aid in this process.

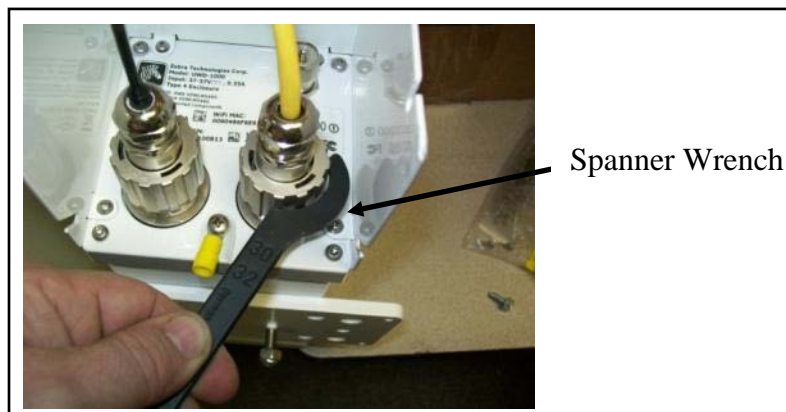


Figure 64: Jack and Plug Engagement

B.2.8 Looping of Ethernet cable. Loop Ethernet cable in a gentle curve and exit "mouse-hole" of drip shield. See Figure .



Figure 24: Looping of Ethernet cable

B.2.9 Unit grounding. Using a **Philip** screwdriver, remove the attached ring terminal from the rear of the unit and crimp onto the user supplied 12 AWG grounding wire. Re-attach to unit using the same screw and washer combo. Hand tighten snugly. See Figure 24 below.

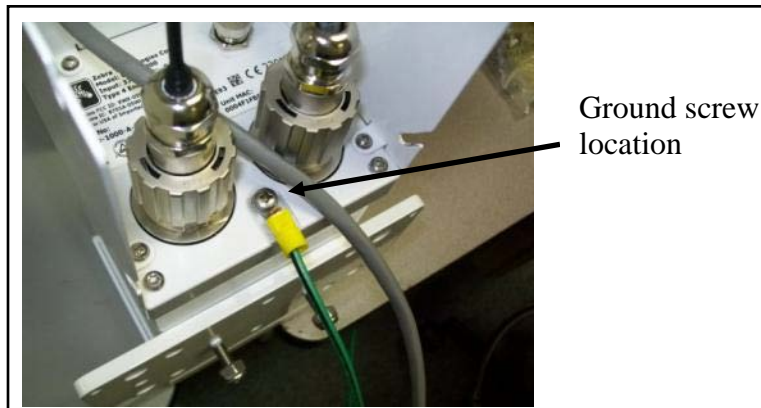


Figure 65: Ground wire attachment

APPENDIX C

Reference: ISO 24730-1 field definitions and descriptions

SLMF_version

Data Type: String

Restriction: 1 to 10 characters

Version of the SLMF implementation, assigned by ISO/IEC 24730. If a draft version of the ISO document is implemented, write as DrftYYMMDD (example, ISO/IEC 2730 Draft from Date 2012/03/19 is Drft120319)

Battery (Mandatory)

Data Type: HexBinary

Restriction: 1byte.

Indicates percentage of battery power remaining, from 0 (0x00 battery dead) to 100 (0x64 – battery at full capacity). For example, a battery with 75% capacity would be expressed as 0x4B (75 decimal). If the battery indicator is not available to the locate message then use value 0xFF.

The mapping from Tag Transmit bits to SLMP is:

Sapphire		Zebra Mode	ISO/IEEE Mode	SLMP_Battery (HexBinary)	Notes
<0x00008000	<0x00200000	≥0x00200000	≥0x0021D000	N/A	Tag ID Range
0x0 – 0x9	0x0 – 0x2	0x0 – 0x9	b01	00	Battery low (0-10%)
N/A	N/A	N/A	b10 ¹	0A	Battery 10% to 30%
0xA - .0xF	0x3 – 0xF	0xA – 0xF	b00	64	Battery good

¹Not implemented by Zebra

Reference: Zebra field definitions and descriptions used by Z-SLMP

Virtual_Group_ID

Data Type: Integer

Restriction: None

Describes the Dart Virtual group ID used to report location.

Sensor_Number

Data Type: HexBinary

Restriction: 1 byte

Local sensor number assigned within the Dart Hub?

Sensor_ID

Data Type: HexBinary

Restriction: 32 bytes

Unique 8 byte ID assigned to a sensor.

Data_Quality_Indication

Data Type: String

Restriction: 1 to 64 characters

Describes the quality of the calculation when additional sensors are used in the computation (see manual). "*" used when data is not available.

Temperature

Data Type: String

Restriction: 1 to 10 characters

Temperature defined in Celsius. "*" used when data is not available

Sensor_List

Data Type: String

Restriction: 1 to 1023 characters

List of sensors that detected event. In format Sxx-Syy-...-Szz where xx,yy, and zz are <Sensor_Number>

Hub_Name

Data Type: String

Restriction: 1 to 255 characters

This value represents the assigned name of the hub or the IP address of the hub.

P_Data

Data Type: String

Restriction: 1 to 64 characters

The reason for the P data (presence data) output. Below are the acceptable values:

Value	Description
P	Only presence data expected. The sensor is not part of a virtual group and is set up to provide only P data.
M<VG#>	Minimum units for the virtual group not met. For example, if the computation for minimum units is set to 3, then less than three sensors detected this tag transmission.
B<VG#>	Bounding Box not met. The system computed the information, but the resulting coordinates were outside of the values defined for the Group Boundary.
C<VG#>	Convergence not met. The system computed the information, but failed to reach a convergence result.
R<VG#>	Reference is currently not available for virtual group computation.
D<VG#>	DQI threshold value for the virtual group exceeded when filtering is enabled.

Event_Message

Data Type: String

Restriction: None

Message associated with an Event_ID for a d-packet. See User Manual's Appendix on D-packets for details.

Event_ID

Data Type: Integer

Restriction: None

Number assigned to an event d-packet. See User Manual's Appendix on D-packets for details.

DVR_Name

Data Type: String

Restriction: Maximum 255 characters

This value represents the DNS name assigned to the DVR.

APPENDIX D: DART VISION READER COUNTRY CODE KEY SETTING

PROBLEM SUMMARY: All Dart Vision Reader with Firmware V5.0.0 (Build Date of **July 10 2012 and later**) will REQUIRE identification of the country of use and Wi-Fi antenna selection **BEFORE** any Wi-Fi operation/connectivity is allowed. Country code and antenna selection are stored as KEYS in the NVRAM. Older version of FW did not have the Country Code KEYS and therefore, immediately subsequent to upgrading to newer FW, all Wi-Fi connectivity will be LOST.

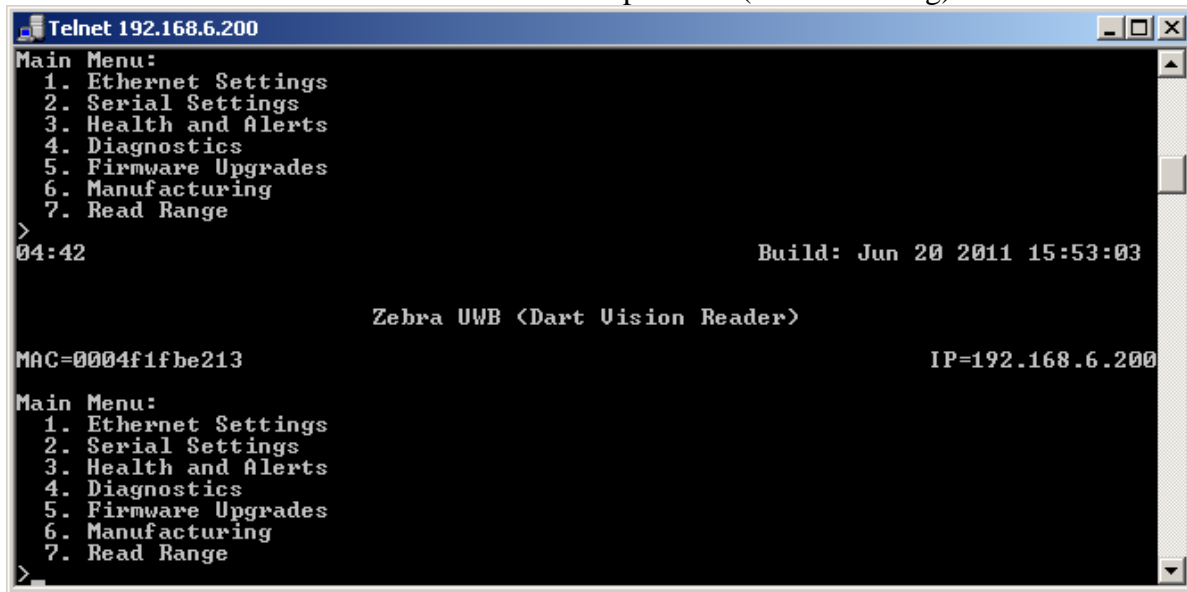
PROBLEM DESCRIPTION: The Dart Vision Reader firmware controls the functionality and power output limits of the Wi-Fi Client Card. If the Country Code is not set, the Wi-Fi Client Card will not be operational.

In Firmware Versions Prior to 5.0.0 Build Date July 10 2012, the firmware did not contain a country code setting. If Dart Vision Reader is wirelessly connected to the network and the unit is upgraded to V5.0.0 or later the Dart Vision Reader will lose wireless connectivity to the network. Then this will require direct connection either with the wired network or serial cable to fix the problem.

FIX: This problem can be avoided by first (prior to FW upgrade) MANUALLY, setting a Country Code Key under the Manufacturing Menu-> Processor Keys-> Add Key-> (add CC for Enter Key, and appropriate Country Code number for Value, for example 840 for the US.) Then ->Write Keys password ff2. It is best to check to make sure the Key was added. >Read Keys then >List Keys. Somewhere in the list, usually bottom, you will see CC=840 for example. It is best to reset the unit, and then upgrade the firmware. See following pages for detailed screen shoots.

Note: Once the unit is upgraded to V5.0.0 July 10 2012 or later there will be a menu item under the Manufacturing Menu to List the Country Codes and Set the Country Code. Contact Professional Services for the latest approved Country Code List.

From Main Menu select item number that corresponds to (Manufacturing)



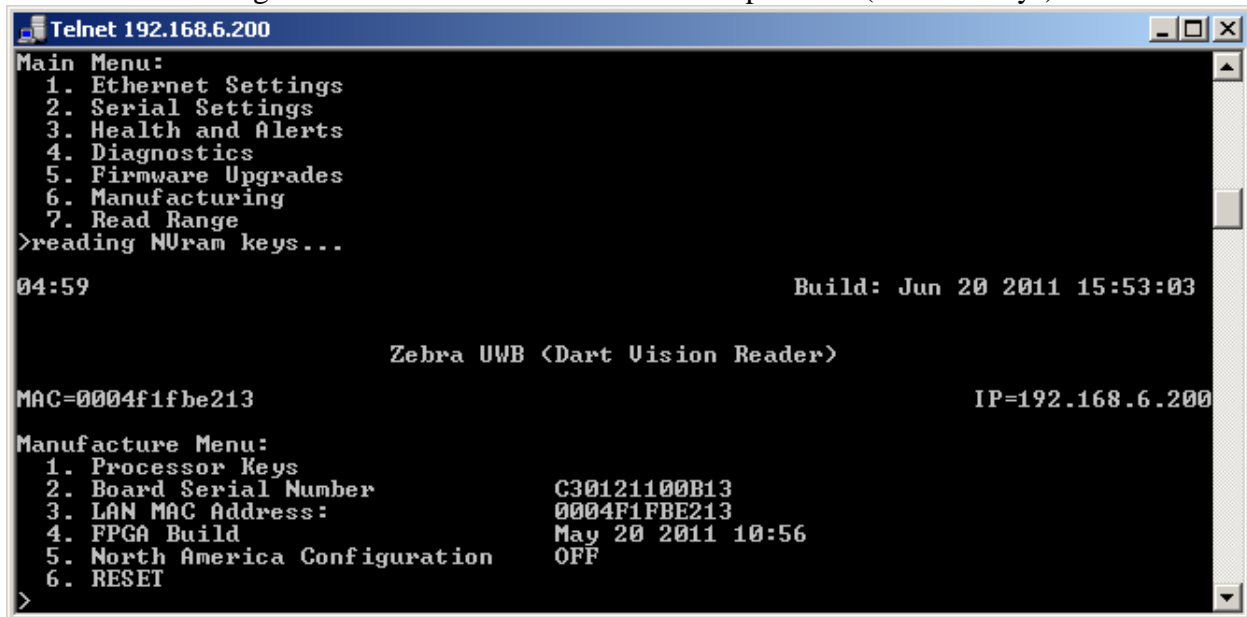
```
Telnet 192.168.6.200
Main Menu:
1. Ethernet Settings
2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>
04:42                               Build: Jun 20 2011 15:53:03

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe213                    IP=192.168.6.200

Main Menu:
1. Ethernet Settings
2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>
```

From Manufacturing Menu select item number that corresponds to (Process Keys)



```
Telnet 192.168.6.200
Main Menu:
1. Ethernet Settings
2. Serial Settings
3. Health and Alerts
4. Diagnostics
5. Firmware Upgrades
6. Manufacturing
7. Read Range
>reading NURam keys...
04:59                               Build: Jun 20 2011 15:53:03

                                Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe213                    IP=192.168.6.200

Manufacture Menu:
1. Processor Keys                    C30121100B13
2. Board Serial Number               0004F1FBE213
3. LAN MAC Address:                  May 20 2011 10:56
4. FPGA Build                        OFF
5. North America Configuration
6. RESET
>
```

From Processor Keys Menu select item number that corresponds to (Add Key)

```

Telnet 192.168.6.200
MAC=0004f1fbe213 IP=192.168.6.200

Manufacture Menu:
1. Processor Keys
2. Board Serial Number      C30121100B13
3. LAN MAC Address:         0004F1FBE213
4. FPGA Build               May 20 2011 10:56
5. North America Configuration OFF
6. RESET
>
05:53 Build: Jun 20 2011 15:53:03

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe213 IP=192.168.6.200

Processor Key Menu:
1. Read Keys
2. List Keys
3. Clear Keys
4. Add Key
5. Delete Key
6. Write Keys
>

```

Add Key for Enter key; CC Enter and For value: appropriate Country Code (Check with Professional Services for Proper Country Code).

```

Telnet 192.168.6.200
3. Clear Keys
4. Add Key
5. Delete Key
6. Write Keys
>To write processor keys, enter password :***
INVALID PASSWORD!

<space> to continue...

06:21 Build: Jun 20 2011 15:53:03

Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe213 IP=192.168.6.200

Processor Key Menu:
1. Read Keys
2. List Keys
3. Clear Keys
4. Add Key
5. Delete Key
6. Write Keys
>
Enter key: CC Enter value: 840_

```

Then select item number for (Write Keys), password is 'ff2'

```

Telnet 192.168.6.200
3. Clear Keys
4. Add Key
5. Delete Key
6. Write Keys
>
Enter key: CC   Enter value: 840
keys 'CC=840' added
<space> to continue...
06:46                               Build: Jun 20 2011 15:53:03

                               Zebra UWB <Dart Vision Reader>

MAC=0004f1fbe213                    IP=192.168.6.200

Processor Key Menu:
1. Read Keys
2. List Keys
3. Clear Keys
4. Add Key
5. Delete Key
6. Write Keys
>To write processor keys, enter password :***

```

Then select (Read Keys) and then (List Keys)

```

Telnet 192.168.6.200
6. Write Keys
>
Key list:
'IPW=192.168.5.200'
'IPL=192.168.6.200'
'ENW=0'
'WPA=1'
'US=0'
'G2=1'
'FM=F'
'BM=F'
'SMW=255.255.255.0'
'DHW=0'
'SML=255.255.255.0'
'ENL=1'
'TP=IFTP'
'UWB=H'
'EA=0004F1FBE213'
'SN=C30121100B13'
'DHL=0'
'TN=1'
'HA=192.168.6.250'
'CC=840'
<space> to continue...

```

Note you will see CC=840 for example.

See next page for list of Country Codes.

Country Code List

Under the Manufacturing Menu of the Dart Vision Reader, there is a place to set the approved country code for the Dart Vision Read, this controls the power output of the Wifi Client Card.

Code	Country
36	Australia
40	Austria
56	Belgium
76	Brazil
100	Bulgaria
124	Canada
196	Cyprus
203	Czech Republic
208	Denmark
818	Egypt
233	Estonia
234	Faeroe Islands
246	Finland
250	France
276	Germany
300	Greece
348	Hungary
352	Iceland
356	India
372	Ireland
380	Italy
414	Kuwait
428	Latvia
438	Liechtenstein
440	Lithuania

Code	Country
442	Luxembourg
470	Malta
484	Mexico
528	Netherlands
554	New Zealand
578	Norway
616	Poland
620	Portugal
642	Romania
643	Russia
702	Singapore
703	Slovakia
705	Slovenia
710	South Africa
724	Spain
752	Sweden
756	Switzerland
764	Thailand
792	Turkey
784	U.A.E.
826	United Kingdom
840	United States

APPENDIX E: WPA_SUPPLICANT SETTINGS DETAILS

The following are a detailed list and explanation of the WPA_Supplicant settings. **For reference only.**

```
##### Example wpa_supplicant configuration file
#####
#
# This file describes configuration file format and lists all available
option.
# Please also take a look at simpler configuration examples in 'examples'
# subdirectory.
#
# Empty lines and lines starting with # are ignored

# NOTE! This file may contain password information and should probably be
made
# readable only by root user on multiuser systems.

# Note: All file paths in this configuration file should use full (absolute,
# not relative to working directory) path in order to allow working directory
# to be changed. This can happen if wpa_supplicant is run in the background.

# Whether to allow wpa_supplicant to update (overwrite) configuration
#
# This option can be used to allow wpa_supplicant to overwrite configuration
# file whenever configuration is changed (e.g., new network block is added
with
# wpa_cli or wpa_gui, or a password is changed). This is required for
# wpa_cli/wpa_gui to be able to store the configuration changes permanently.
# Please note that overwriting configuration file will remove the comments
from
# it.
#update_config=1

# global configuration (shared by all network blocks)
#
# Parameters for the control interface. If this is specified, wpa_supplicant
# will open a control interface that is available for external programs to
# manage wpa_supplicant. The meaning of this string depends on which control
# interface mechanism is used. For all cases, the existence of this parameter
# in configuration is used to determine whether the control interface is
# enabled.
#
# For UNIX domain sockets (default on Linux and BSD): This is a directory
that
# will be created for UNIX domain sockets for listening to requests from
# external programs (CLI/GUI, etc.) for status information and configuration.
# The socket file will be named based on the interface name, so multiple
# wpa_supplicant processes can be run at the same time if more than one
# interface is used.
# /var/run/wpa_supplicant is the recommended directory for sockets and by
# default, wpa_cli will use it when trying to connect with wpa_supplicant.
#
# Access control for the control interface can be configured by setting the
# directory to allow only members of a group to use sockets. This way, it is
```



```
# possible to run wpa_supplicant as root (since it needs to change network
# configuration and open raw sockets) and still allow GUI/CLI components to
# be
# run as non-root users. However, since the control interface can be used to
# change the network configuration, this access needs to be protected in many
# cases. By default, wpa_supplicant is configured to use gid 0 (root). If you
# want to allow non-root users to use the control interface, add a new group
# and change this value to match with that group. Add users that should have
# control interface access to this group. If this variable is commented out
# or
# not included in the configuration file, group will not be changed from the
# value it got by default when the directory or socket was created.
#
# When configuring both the directory and group, use following format:
# DIR=/var/run/wpa_supplicant GROUP=wheel
# DIR=/var/run/wpa_supplicant GROUP=0
# (group can be either group name or gid)
#
# For UDP connections (default on Windows): The value will be ignored. This
# variable is just used to select that the control interface is to be
# created.
# The value can be set to, e.g., udp (ctrl_interface=udp)
#
# For Windows Named Pipe: This value can be used to set the security
# descriptor
# for controlling access to the control interface. Security descriptor can be
# set using Security Descriptor String Format (see http://msdn.microsoft.com/
# library/default.asp?url=/library/en-us/secauthz/security/
# security_descriptor_string_format.asp). The descriptor string needs to be
# prefixed with SDDL=. For example, ctrl_interface=SDDL=D: would set an empty
# DACL (which will reject all connections). See README-Windows.txt for more
# information about SDDL string format.
#
ctrl_interface=/var/run/wpa_supplicant

# IEEE 802.1X/EAPOL version
# wpa_supplicant is implemented based on IEEE Std 802.1X-2004 which defines
# EAPOL version 2. However, there are many APs that do not handle the new
# version number correctly (they seem to drop the frames completely). In
# order
# to make wpa_supplicant interoperate with these APs, the version number is
# set
# to 1 by default. This configuration value can be used to set it to the new
# version (2).
eapol_version=1

# AP scanning/selection
# By default, wpa_supplicant requests driver to perform AP scanning and then
# uses the scan results to select a suitable AP. Another alternative is to
# allow the driver to take care of AP scanning and selection and use
# wpa_supplicant just to process EAPOL frames based on IEEE 802.11
# association
# information from the driver.
# 1: wpa_supplicant initiates scanning and AP selection; if no APs matching
# to
# the currently enabled networks are found, a new network (IBSS or AP mode
# operation) may be initialized (if configured) (default)
```

```
# 0: driver takes care of scanning, AP selection, and IEEE 802.11 association
#   parameters (e.g., WPA IE generation); this mode can also be used with
#   non-WPA drivers when using IEEE 802.1X mode; do not try to associate
with
#   APs (i.e., external program needs to control association). This mode
must
#   also be used when using wired Ethernet drivers.
# 2: like 0, but associate with APs using security policy and SSID (but not
#   BSSID); this can be used, e.g., with ndiswrapper and NDIS drivers to
#   enable operation with hidden SSIDs and optimized roaming; in this mode,
#   the network blocks in the configuration file are tried one by one until
#   the driver reports successful association; each network block should
have
#   explicit security policy (i.e., only one option in the lists) for
#   key_mgmt, pairwise, group, proto variables
# When using IBSS or AP mode, ap_scan=2 mode can force the new network to be
# created immediately regardless of scan results. ap_scan=1 mode will first
try
# to scan for existing networks and only if no matches with the enabled
# networks are found, a new IBSS or AP mode network is created.
ap_scan=1

# EAP fast re-authentication
# By default, fast re-authentication is enabled for all EAP methods that
# support it. This variable can be used to disable fast re-authentication.
# Normally, there is no need to disable this.
fast_reauth=1

# OpenSSL Engine support
# These options can be used to load OpenSSL engines.
# The two engines that are supported currently are shown below:
# They are both from the openssl project (http://www.openssl.org/)
# By default no engines are loaded.
# make the openssl engine available
#openssl_engine_path=/usr/lib/openssl/engine_openssl.so
# make the pkcs11 engine available
#pkcs11_engine_path=/usr/lib/openssl/engine_pkcs11.so
# configure the path to the pkcs11 module required by the pkcs11 engine
#pkcs11_module_path=/usr/lib/pkcs11/openssl-pkcs11.so

# Dynamic EAP methods
# If EAP methods were built dynamically as shared object files, they need to
be
# loaded here before being used in the network blocks. By default, EAP
methods
# are included statically in the build, so these lines are not needed
#load_dynamic_eap=/usr/lib/wpa_supplicant/eap_tls.so
#load_dynamic_eap=/usr/lib/wpa_supplicant/eap_md5.so

# Driver interface parameters
# This field can be used to configure arbitrary driver interface parameters.
The
# format is specific to the selected driver interface. This field is not used
# in most cases.
#driver_param="field=value"

# Country code
```

```
# The ISO/IEC alpha2 country code for the country in which this device is
# currently operating.
#country=US

# Maximum lifetime for PMKSA in seconds; default 43200
#dot11RSNAConfigPMKLifetime=43200
# Threshold for reauthentication (percentage of PMK lifetime); default 70
#dot11RSNAConfigPMKReauthThreshold=70
# Timeout for security association negotiation in seconds; default 60
#dot11RSNAConfigSATimeout=60

# Wi-Fi Protected Setup (WPS) parameters

# Universally Unique IDentifier (UUID; see RFC 4122) of the device
# If not configured, UUID will be generated based on the local MAC address.
#uuid=12345678-9abc-def0-1234-56789abcdef0

# Device Name
# User-friendly description of device; up to 32 octets encoded in UTF-8
#device_name=Wireless Client

# Manufacturer
# The manufacturer of the device (up to 64 ASCII characters)
#manufacturer=Company

# Model Name
# Model of the device (up to 32 ASCII characters)
#model_name=cmodel

# Model Number
# Additional device description (up to 32 ASCII characters)
#model_number=123

# Serial Number
# Serial number of the device (up to 32 characters)
#serial_number=12345

# Primary Device Type
# Used format: <categ>-<OUI>-<subcateg>
# categ = Category as an integer value
# OUI = OUI and type octet as a 4-octet hex-encoded value; 0050F204 for
#         default WPS OUI
# subcateg = OUI-specific Sub Category as an integer value
# Examples:
#   1-0050F204-1 (Computer / PC)
#   1-0050F204-2 (Computer / Server)
#   5-0050F204-1 (Storage / NAS)
#   6-0050F204-1 (Network Infrastructure / AP)
#device_type=1-0050F204-1

# OS Version
# 4-octet operating system version number (hex string)
#os_version=01020300

# Config Methods
# List of the supported configuration methods
# Available methods: usba ethernet label display ext_nfc_token int_nfc_token
```

```
#      nfc_interface push_button keypad virtual_display physical_display
#      virtual_push_button physical_push_button
# For WSC 1.0:
#config_methods=label display push_button keypad
# For WSC 2.0:
#config_methods=label virtual_display virtual_push_button keypad

# Credential processing
# 0 = process received credentials internally (default)
# 1 = do not process received credentials; just pass them over ctrl_iface
to
#      external program(s)
# 2 = process received credentials internally and pass them over ctrl_iface
#      to external program(s)
#wps_cred_processing=0

# Vendor attribute in WPS M1, e.g., Windows 7 Vertical Pairing
# The vendor attribute contents to be added in M1 (hex string)
#wps_vendor_ext_m1=000137100100020001

# NFC password token for WPS
# These parameters can be used to configure a fixed NFC password token for
the
# station. This can be generated, e.g., with nfc_pw_token. When these
# parameters are used, the station is assumed to be deployed with a NFC tag
# that includes the matching NFC password token (e.g., written based on the
# NDEF record from nfc_pw_token).
#
#wps_nfc_dev_pw_id: Device Password ID (16..65535)
#wps_nfc_dh_pubkey: Hexdump of DH Public Key
#wps_nfc_dh_privkey: Hexdump of DH Private Key
#wps_nfc_dev_pw: Hexdump of Device Password

# Maximum number of BSS entries to keep in memory
# Default: 200
# This can be used to limit memory use on the BSS entries (cached scan
# results). A larger value may be needed in environments that have huge
number
# of APs when using ap_scan=1 mode.
#bss_max_count=200

# Automatic scan
# This is an optional set of parameters for automatic scanning
# within an interface in following format:
#autoscan=<autoscan module name>:<module parameters>
#Â autoscan is like bgscan but on disconnected or inactive state.
#Â For instance, on exponential module parameters would be <base>:<limit>
#autoscan=exponential:3:300
# Which means a delay between scans on a base exponential of 3,
#Â up to the limit of 300 seconds (3, 9, 27 ... 300)
#Â For periodic module, parameters would be <fixed interval>
#autoscan=periodic:30
#Â So a delay of 30 seconds will be applied between each scan

# filter_ssids - SSID-based scan result filtering
# 0 = do not filter scan results (default)
# 1 = only include configured SSIDs in scan results/BSS table
```

```
#filter_ssids=0

# Password (and passphrase, etc.) backend for external storage
# format: <backend name>[:<optional backend parameters>]
#ext_password_backend=test:pw1=password|pw2=testing

# Timeout in seconds to detect STA inactivity (default: 300 seconds)
#
# This timeout value is used in P2P GO mode to clean up
# inactive stations.
#p2p_go_max_inactivity=300

# Opportunistic Key Caching (also known as Proactive Key Caching) default
# This parameter can be used to set the default behavior for the
# proactive_key_caching parameter. By default, OKC is disabled unless enabled
# with the global okc=1 parameter or with the per-network
# proactive_key_caching=1 parameter. With okc=1, OKC is enabled by default,
# but
# can be disabled with per-network proactive_key_caching=0 parameter.
#okc=0

# Protected Management Frames default
# This parameter can be used to set the default behavior for the ieee80211w
# parameter. By default, PMF is disabled unless enabled with the global
# pmf=1/2
# parameter or with the per-network ieee80211w=1/2 parameter. With pmf=1/2,
# PMF
# is enabled/required by default, but can be disabled with the per-network
# ieee80211w parameter.
#pmf=0

# Enabled SAE finite cyclic groups in preference order
# By default (if this parameter is not set), the mandatory group 19 (ECC
# group
# defined over a 256-bit prime order field) is preferred, but other groups
# are
# also enabled. If this parameter is set, the groups will be tried in the
# indicated order. The group values are listed in the IANA registry:
# http://www.iana.org/assignments/ipsec-registry/ipsec-registry.xml#ipsec-registry-9
#sae_groups=21 20 19 26 25

# Default value for DTIM period (if not overridden in network block)
#dtim_period=2

# Default value for Beacon interval (if not overridden in network block)
#beacon_int=100

# Additional vendor specific elements for Beacon and Probe Response frames
# This parameter can be used to add additional vendor specific element(s)
# into
# the end of the Beacon and Probe Response frames. The format for these
# element(s) is a hexdump of the raw information elements (id+len+payload for
# one or more elements). This is used in AP and P2P GO modes.
#ap_vendor_elements=dd0411223301

# Ignore scan results older than request
```

```
#
# The driver may have a cache of scan results that makes it return
# information that is older than our scan trigger. This parameter can
# be used to configure such old information to be ignored instead of
# allowing it to update the internal BSS table.
#ignore_old_scan_res=0

# scan_cur_freq: Whether to scan only the current frequency
# 0: Scan all available frequencies. (Default)
# 1: Scan current operating frequency if another VIF on the same radio
#     is already associated.

# Interworking (IEEE 802.11u)

# Enable Interworking
# interworking=1

# Homogenous ESS identifier
# If this is set, scans will be used to request response only from BSSes
# belonging to the specified Homogeneous ESS. This is used only if
interworking
# is enabled.
# hessid=00:11:22:33:44:55

# Automatic network selection behavior
# 0 = do not automatically go through Interworking network selection
#     (i.e., require explicit interworking_select command for this; default)
# 1 = perform Interworking network selection if one or more
#     credentials have been configured and scan did not find a
#     matching network block
#auto_interworking=0

# credential block
#
# Each credential used for automatic network selection is configured as a set
# of parameters that are compared to the information advertised by the APs
when
# interworking_select and interworking_connect commands are used.
#
# credential fields:
#
# priority: Priority group
#     By default, all networks and credentials get the same priority group
#     (0). This field can be used to give higher priority for credentials
#     (and similarly in struct wpa_ssid for network blocks) to change the
#     Interworking automatic networking selection behavior. The matching
#     network (based on either an enabled network block or a credential)
#     with the highest priority value will be selected.
#
# pcsc: Use PC/SC and SIM/USIM card
#
# realm: Home Realm for Interworking
#
# username: Username for Interworking network selection
#
# password: Password for Interworking network selection
#
```

```
# ca_cert: CA certificate for Interworking network selection
#
# client_cert: File path to client certificate file (PEM/DER)
#     This field is used with Interworking networking selection for a case
#     where client certificate/private key is used for authentication
#     (EAP-TLS). Full path to the file should be used since working
#     directory may change when wpa_supplicant is run in the background.
#
#     Alternatively, a named configuration blob can be used by setting
#     this to blob://blob_name.
#
# private_key: File path to client private key file (PEM/DER/PFX)
#     When PKCS#12/PFX file (.p12/.pfx) is used, client_cert should be
#     commented out. Both the private key and certificate will be read
#     from the PKCS#12 file in this case. Full path to the file should be
#     used since working directory may change when wpa_supplicant is run
#     in the background.
#
#     Windows certificate store can be used by leaving client_cert out and
#     configuring private_key in one of the following formats:
#
#     cert://substring_to_match
#
#     hash://certificate_thumbprint_in_hex
#
#     For example: private_key="hash://63093aa9c47f56ae88334c7b65a4"
#
#     Note that when running wpa_supplicant as an application, the user
#     certificate store (My user account) is used, whereas computer store
#     (Computer account) is used when running wpasvc as a service.
#
#     Alternatively, a named configuration blob can be used by setting
#     this to blob://blob_name.
#
# private_key_passwd: Password for private key file
#
# imsi: IMSI in <MCC> | <MNC> | '-' | <MSIN> format
#
# milenage: Milenage parameters for SIM/USIM simulator in <Ki>:<OPc>:<SQN>
#     format
#
# domain: Home service provider FQDN
#     This is used to compare against the Domain Name List to figure out
#     whether the AP is operated by the Home SP.
#
# roaming_consortium: Roaming Consortium OI
#     If roaming_consortium_len is non-zero, this field contains the
#     Roaming Consortium OI that can be used to determine which access
#     points support authentication with this credential. This is an
#     alternative to the use of the realm parameter. When using Roaming
#     Consortium to match the network, the EAP parameters need to be
#     pre-configured with the credential since the NAI Realm information
#     may not be available or fetched.
#
# eap: Pre-configured EAP method
#     This optional field can be used to specify which EAP method will be
#     used with this credential. If not set, the EAP method is selected
```

```
#         automatically based on ANQP information (e.g., NAI Realm).
#
# phase1: Pre-configure Phase 1 (outer authentication) parameters
#         This optional field is used with like the 'eap' parameter.
#
# phase2: Pre-configure Phase 2 (inner authentication) parameters
#         This optional field is used with like the 'eap' parameter.
#
# excluded_ssid: Excluded SSID
#         This optional field can be used to excluded specific SSID(s) from
#         matching with the network. Multiple entries can be used to specify
more
#         than one SSID.
#
# for example:
#
# cred={
#     realm="example.com"
#     username="user@example.com"
#     password="password"
#     ca_cert="/etc/wpa_supplicant/ca.pem"
#     domain="example.com"
# }
#
# cred={
#     imsi="310026-0000000000"
#
#     milenage="90dca4eda45b53cf0f12d7c9c3bc6a89:cb9cccc4b9258e6dca4760379fb
82"
# }
#
# cred={
#     realm="example.com"
#     username="user"
#     password="password"
#     ca_cert="/etc/wpa_supplicant/ca.pem"
#     domain="example.com"
#     roaming_consortium=223344
#     eap=TTLS
#     phase2="auth=MSCHAPV2"
# }

# Hotspot 2.0
# hs20=1

# network block
#
# Each network (usually AP's sharing the same SSID) is configured as a
# separate
# block in this configuration file. The network blocks are in preference
# order
# (the first match is used).
#
# network block fields:
#
# disabled:
#     0 = this network can be used (default)
```



```
#      1 = this network block is disabled (can be enabled through ctrl_iface,
#      e.g., with wpa_cli or wpa_gui)
#
# id_str: Network identifier string for external scripts. This value is
passed
#      to external action script through wpa_cli as WPA_ID_STR environment
#      variable to make it easier to do network specific configuration.
#
# ssid: SSID (mandatory); network name in one of the optional formats:
#      - an ASCII string with double quotation
#      - a hex string (two characters per octet of SSID)
#      - a printf-escaped ASCII string P"<escaped string>"
#
# scan_ssid:
#      0 = do not scan this SSID with specific Probe Request frames (default)
#      1 = scan with SSID-specific Probe Request frames (this can be used to
#          find APs that do not accept broadcast SSID or use multiple SSIDs;
#          this will add latency to scanning, so enable this only when
needed)
#
# bssid: BSSID (optional); if set, this network block is used only when
#      associating with the AP using the configured BSSID
#
# priority: priority group (integer)
# By default, all networks will get same priority group (0). If some of the
# networks are more desirable, this field can be used to change the order in
# which wpa_supplicant goes through the networks when selecting a BSS. The
# priority groups will be iterated in decreasing priority (i.e., the larger the
# priority value, the sooner the network is matched against the scan
results).
# Within each priority group, networks will be selected based on security
# policy, signal strength, etc.
# Please note that AP scanning with scan_ssid=1 and ap_scan=2 mode are not
# using this priority to select the order for scanning. Instead, they try the
# networks in the order that used in the configuration file.
#
# mode: IEEE 802.11 operation mode
# 0 = infrastructure (Managed) mode, i.e., associate with an AP (default)
# 1 = IBSS (ad-hoc, peer-to-peer)
# 2 = AP (access point)
# Note: IBSS can only be used with key_mgmt NONE (plaintext and static WEP)
# and key_mgmt=WPA-NONE (fixed group key TKIP/CCMP). WPA-None requires
# following network block options:
# proto=WPA, key_mgmt=WPA-NONE, pairwise=NONE, group=TKIP (or CCMP, but not
# both), and psk must also be set.
#
# frequency: Channel frequency in megahertz (MHz) for IBSS, e.g.,
# 2412 = IEEE 802.11b/g channel 1. This value is used to configure the
initial
# channel for IBSS (adhoc) networks. It is ignored in the infrastructure
mode.
# In addition, this value is only used by the station that creates the IBSS.
If
# an IBSS network with the configured SSID is already present, the frequency
of
# the network will be used instead of this configured value.
```

```
#
# scan_freq: List of frequencies to scan
# Space-separated list of frequencies in MHz to scan when searching for this
# BSS. If the subset of channels used by the network is known, this option
# can
# be used to optimize scanning to not occur on channels that the network does
# not use. Example: scan_freq=2412 2437 2462
#
# freq_list: Array of allowed frequencies
# Space-separated list of frequencies in MHz to allow for selecting the BSS.
# If
# set, scan results that do not match any of the specified frequencies are
# not
# considered when selecting a BSS.
#
# This can also be set on the outside of the network block. In this case,
# it limits the frequencies that will be scanned.
#
# bgscan: Background scanning
# wpa_supplicant behavior for background scanning can be specified by
# configuring a bgscan module. These modules are responsible for requesting
# background scans for the purpose of roaming within an ESS (i.e., within a
# single network block with all the APs using the same SSID). The bgscan
# parameter uses following format: "<bgscan module name>:<module parameters>"
# Following bgscan modules are available:
# simple - Periodic background scans based on signal strength
# bgscan="simple:<short bgscan interval in seconds>:<signal strength
# threshold>:
# <long interval>"
# bgscan="simple:30:-45:300"
# learn - Learn channels used by the network and try to avoid bgscans on
# other
# channels (experimental)
# bgscan="learn:<short bgscan interval in seconds>:<signal strength
# threshold>:
# <long interval>[:<database file name>]"
# bgscan="learn:30:-45:300:/etc/wpa_supplicant/network1.bgscan"
#
# proto: list of accepted protocols
# WPA = WPA/IEEE 802.11i/D3.0
# RSN = WPA2/IEEE 802.11i (also WPA2 can be used as an alias for RSN)
# If not set, this defaults to: WPA RSN
#
# key_mgmt: list of accepted authenticated key management protocols
# WPA-PSK = WPA pre-shared key (this requires 'psk' field)
# WPA-EAP = WPA using EAP authentication
# IEEE8021X = IEEE 802.1X using EAP authentication and (optionally)
# dynamically
# generated WEP keys
# NONE = WPA is not used; plaintext or static WEP could be used
# WPA-PSK-SHA256 = Like WPA-PSK but using stronger SHA256-based algorithms
# WPA-EAP-SHA256 = Like WPA-EAP but using stronger SHA256-based algorithms
# If not set, this defaults to: WPA-PSK WPA-EAP
#
# ieee80211w: whether management frame protection is enabled
# 0 = disabled (default unless changed with the global pmf parameter)
# 1 = optional
```

```
# 2 = required
# The most common configuration options for this based on the PMF (protected
# management frames) certification program are:
# PMF enabled: ieee80211w=1 and key_mgmt=WPA-EAP WPA-EAP-SHA256
# PMF required: ieee80211w=2 and key_mgmt=WPA-EAP-SHA256
# (and similarly for WPA-PSK and WPA-WPSK-SHA256 if WPA2-Personal is used)
#
# auth_alg: list of allowed IEEE 802.11 authentication algorithms
# OPEN = Open System authentication (required for WPA/WPA2)
# SHARED = Shared Key authentication (requires static WEP keys)
# LEAP = LEAP/Network EAP (only used with LEAP)
# If not set, automatic selection is used (Open System with LEAP enabled if
# LEAP is allowed as one of the EAP methods).
#
# pairwise: list of accepted pairwise (unicast) ciphers for WPA
# CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0]
# TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0]
# NONE = Use only Group Keys (deprecated, should not be included if APs
support
# pairwise keys)
# If not set, this defaults to: CCMP TKIP
#
# group: list of accepted group (broadcast/multicast) ciphers for WPA
# CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i/D7.0]
# TKIP = Temporal Key Integrity Protocol [IEEE 802.11i/D7.0]
# WEP104 = WEP (Wired Equivalent Privacy) with 104-bit key
# WEP40 = WEP (Wired Equivalent Privacy) with 40-bit key [IEEE 802.11]
# If not set, this defaults to: CCMP TKIP WEP104 WEP40
#
# psk: WPA preshared key; 256-bit pre-shared key
# The key used in WPA-PSK mode can be entered either as 64 hex-digits, i.e.,
# 32 bytes or as an ASCII passphrase (in which case, the real PSK will be
# generated using the passphrase and SSID). ASCII passphrase must be between
# 8 and 63 characters (inclusive). ext:<name of external PSK field> format
can
# be used to indicate that the PSK/passphrase is stored in external storage.
# This field is not needed, if WPA-EAP is used.
# Note: Separate tool, wpa_passphrase, can be used to generate 256-bit keys
# from ASCII passphrase. This process uses lot of CPU and wpa_supplicant
# startup and reconfiguration time can be optimized by generating the PSK
only
# only when the passphrase or SSID has actually changed.
#
# eapol_flags: IEEE 802.1X/EAPOL options (bit field)
# Dynamic WEP key required for non-WPA mode
# bit0 (1): require dynamically generated unicast WEP key
# bit1 (2): require dynamically generated broadcast WEP key
# (3 = require both keys; default)
# Note: When using wired authentication, eapol_flags must be set to 0 for the
# authentication to be completed successfully.
#
# mixed_cell: This option can be used to configure whether so called mixed
# cells, i.e., networks that use both plaintext and encryption in the same
# SSID, are allowed when selecting a BSS from scan results.
# 0 = disabled (default)
# 1 = enabled
#
```

```
# proactive_key_caching:
# Enable/disable opportunistic PMKSA caching for WPA2.
# 0 = disabled (default unless changed with the global okc parameter)
# 1 = enabled
#
# wep_key0..3: Static WEP key (ASCII in double quotation, e.g. "abcde" or
# hex without quotation, e.g., 0102030405)
# wep_tx_keyidx: Default WEP key index (TX) (0..3)
#
# peerkey: Whether PeerKey negotiation for direct links (IEEE 802.11e DLS) is
# allowed. This is only used with RSN/WPA2.
# 0 = disabled (default)
# 1 = enabled
#peerkey=1
#
# wpa_ptk_rekey: Maximum lifetime for PTK in seconds. This can be used to
# enforce rekeying of PTK to mitigate some attacks against TKIP deficiencies.
#
# Following fields are only used with internal EAP implementation.
# eap: space-separated list of accepted EAP methods
#     MD5 = EAP-MD5 (unsecure and does not generate keying material ->
#                   cannot be used with WPA; to be used as a Phase 2 method
#                   with EAP-PEAP or EAP-TTLS)
#     MSCHAPV2 = EAP-MSCHAPv2 (cannot be used separately with WPA; to be
used
#                   as a Phase 2 method with EAP-PEAP or EAP-TTLS)
#     OTP = EAP-OTP (cannot be used separately with WPA; to be used
#                   as a Phase 2 method with EAP-PEAP or EAP-TTLS)
#     GTC = EAP-GTC (cannot be used separately with WPA; to be used
#                   as a Phase 2 method with EAP-PEAP or EAP-TTLS)
#     TLS = EAP-TLS (client and server certificate)
#     PEAP = EAP-PEAP (with tunnelled EAP authentication)
#     TTLS = EAP-TTLS (with tunnelled EAP or PAP/CHAP/MSCHAP/MSCHAPV2
#                   authentication)
#     If not set, all compiled in methods are allowed.
#
# identity: Identity string for EAP
#     This field is also used to configure user NAI for
#     EAP-PSK/PAX/SAKE/GPSK.
# anonymous_identity: Anonymous identity string for EAP (to be used as the
# unencrypted identity with EAP types that support different tunnelled
# identity, e.g., EAP-TTLS). This field can also be used with
# EAP-SIM/AKA/AKA' to store the pseudonym identity.
# password: Password string for EAP. This field can include either the
# plaintext password (using ASCII or hex string) or a NtPasswordHash
# (16-byte MD4 hash of password) in hash:<32 hex digits> format.
# NtPasswordHash can only be used when the password is for MSCHAPv2 or
# MSCHAP (EAP-MSCHAPv2, EAP-TTLS/MSCHAPv2, EAP-TTLS/MSCHAP, LEAP).
# EAP-PSK (128-bit PSK), EAP-PAX (128-bit PSK), and EAP-SAKE (256-bit
# PSK) is also configured using this field. For EAP-GPSK, this is a
# variable length PSK. ext:<name of external password field> format can
# be used to indicate that the password is stored in external storage.
# ca_cert: File path to CA certificate file (PEM/DER). This file can have one
# or more trusted CA certificates. If ca_cert and ca_path are not
# included, server certificate will not be verified. This is insecure
and
#     a trusted CA certificate should always be configured when using
```

```

#      EAP-TLS/TTLS/PEAP. Full path should be used since working directory
may
#      change when wpa_supplicant is run in the background.
#
#      Alternatively, this can be used to only perform matching of the server
#      certificate (SHA-256 hash of the DER encoded X.509 certificate). In
#      this case, the possible CA certificates in the server certificate
chain
#      are ignored and only the server certificate is verified. This is
#      configured with the following format:
#      hash://server/sha256/cert_hash_in_hex
#      For example: "hash://server/sha256/
#      5albc1296205e6fdb3979728efe3920798885c1c4590b5f90f43222d239ca6a"
#
#      On Windows, trusted CA certificates can be loaded from the system
#      certificate store by setting this to cert_store://<name>, e.g.,
#      ca_cert="cert_store://CA" or ca_cert="cert_store://ROOT".
#      Note that when running wpa_supplicant as an application, the user
#      certificate store (My user account) is used, whereas computer store
#      (Computer account) is used when running wpasvc as a service.
# ca_path: Directory path for CA certificate files (PEM). This path may
#      contain multiple CA certificates in OpenSSL format. Common use for
this
#      is to point to system trusted CA list which is often installed into
#      directory like /etc/ssl/certs. If configured, these certificates are
#      added to the list of trusted CAs. ca_cert may also be included in that
#      case, but it is not required.
# client_cert: File path to client certificate file (PEM/DER)
#      Full path should be used since working directory may change when
#      wpa_supplicant is run in the background.
#      Alternatively, a named configuration blob can be used by setting this
#      to blob://<blob name>.
# private_key: File path to client private key file (PEM/DER/PFX)
#      When PKCS#12/PFX file (.p12/.pfx) is used, client_cert should be
#      commented out. Both the private key and certificate will be read from
#      the PKCS#12 file in this case. Full path should be used since working
#      directory may change when wpa_supplicant is run in the background.
#      Windows certificate store can be used by leaving client_cert out and
#      configuring private_key in one of the following formats:
#      cert://substring_to_match
#      hash://certificate_thumbprint_in_hex
#      for example: private_key="hash://63093aa9c47f56ae88334c7b65a4"
#      Note that when running wpa_supplicant as an application, the user
#      certificate store (My user account) is used, whereas computer store
#      (Computer account) is used when running wpasvc as a service.
#      Alternatively, a named configuration blob can be used by setting this
#      to blob://<blob name>.
# private_key_passwd: Password for private key file (if left out, this will
be
#      asked through control interface)
# dh_file: File path to DH/DSA parameters file (in PEM format)
#      This is an optional configuration file for setting parameters for an
#      ephemeral DH key exchange. In most cases, the default RSA
#      authentication does not use this configuration. However, it is
possible
#      setup RSA to use ephemeral DH key exchange. In addition, ciphers with
#      DSA keys always use ephemeral DH keys. This can be used to achieve

```

```

#         forward secrecy. If the file is in DSA parameters format, it will be
#         automatically converted into DH params.
# subject_match: Substring to be matched against the subject of the
#         authentication server certificate. If this string is set, the server
#         certificate is only accepted if it contains this string in the
subject.
#         The subject string is in following format:
#         /C=US/ST=CA/L=San Francisco/CN=Test AS/emailAddress=as@example.com
# altsubject_match: Semicolon separated string of entries to be matched
against
#         the alternative subject name of the authentication server certificate.
#         If this string is set, the server certificate is only accepted if it
#         contains one of the entries in an alternative subject name extension.
#         altSubjectName string is in following format: TYPE:VALUE
#         Example: EMAIL:server@example.com
#         Example: DNS:server.example.com;DNS:server2.example.com
#         Following types are supported: EMAIL, DNS, URI
# phase1: Phase1 (outer authentication, i.e., TLS tunnel) parameters
#         (string with field-value pairs, e.g., "peapver=0" or
#         "peapver=1 peaplabel=1")
#         'peapver' can be used to force which PEAP version (0 or 1) is used.
#         'peaplabel=1' can be used to force new label, "client PEAP
encryption",
#         to be used during key derivation when PEAPv1 or newer. Most existing
#         PEAPv1 implementation seem to be using the old label, "client EAP
#         encryption", and wpa_supplicant is now using that as the default
value.
#         Some servers, e.g., Radiator, may require peaplabel=1 configuration to
#         interoperate with PEAPv1; see eap_testing.txt for more details.
#         'peap_outer_success=0' can be used to terminate PEAP authentication on
#         tunneled EAP-Success. This is required with some RADIUS servers that
#         implement draft-josefsson-pppext-eap-tls-eap-05.txt (e.g.,
#         Lucent NavisRadius v4.4.0 with PEAP in "IETF Draft 5" mode)
#         include_tls_length=1 can be used to force wpa_supplicant to include
#         TLS Message Length field in all TLS messages even if they are not
#         fragmented.
#         sim_min_num_chal=3 can be used to configure EAP-SIM to require three
#         challenges (by default, it accepts 2 or 3)
#         result_ind=1 can be used to enable EAP-SIM and EAP-AKA to use
#         protected result indication.
#         'crypto_binding' option can be used to control PEAPv0 cryptobinding
#         behavior:
#         * 0 = do not use cryptobinding (default)
#         * 1 = use cryptobinding if server supports it
#         * 2 = require cryptobinding
#         EAP-WSC (WPS) uses following options: pin=<Device Password> or
#         pbc=1.
# phase2: Phase2 (inner authentication with TLS tunnel) parameters
#         (string with field-value pairs, e.g., "auth=MSCHAPV2" for EAP-PEAP or
#         "autheap=MSCHAPV2 autheap=MD5" for EAP-TTLS)
#
# TLS-based methods can use the following parameters to control TLS behavior
# (these are normally in the phase1 parameter, but can be used also in the
# phase2 parameter when EAP-TLS is used within the inner tunnel):
# tls_allow_md5=1 - allow MD5-based certificate signatures (depending on the
#         TLS library, these may be disabled by default to enforce stronger
#         security)

```

```
# tls_disable_time_checks=1 - ignore certificate validity time (this requests
#   the TLS library to accept certificates even if they are not currently
#   valid, i.e., have expired or have not yet become valid; this should be
#   used only for testing purposes)
# tls_disable_session_ticket=1 - disable TLS Session Ticket extension
# tls_disable_session_ticket=0 - allow TLS Session Ticket extension to be
used
#   Note: If not set, this is automatically set to 1 for EAP-TLS/PEAP/TTLS
#   as a workaround for broken authentication server implementations
unless
#   EAP workarounds are disabled with eap_workarounds=0.
#   For EAP-FAST, this must be set to 0 (or left unconfigured for the
#   default value to be used automatically).
#
# Following certificate/private key fields are used in inner Phase2
# authentication when using EAP-TTLS or EAP-PEAP.
# ca_cert2: File path to CA certificate file. This file can have one or more
#   trusted CA certificates. If ca_cert2 and ca_path2 are not included,
#   server certificate will not be verified. This is insecure and a
trusted
#   CA certificate should always be configured.
# ca_path2: Directory path for CA certificate files (PEM)
# client_cert2: File path to client certificate file
# private_key2: File path to client private key file
# private_key2_passwd: Password for private key file
# dh_file2: File path to DH/DSA parameters file (in PEM format)
# subject_match2: Substring to be matched against the subject of the
#   authentication server certificate.
# altsubject_match2: Substring to be matched against the alternative subject
#   name of the authentication server certificate.
#
# fragment_size: Maximum EAP fragment size in bytes (default 1398).
#   This value limits the fragment size for EAP methods that support
#   fragmentation (e.g., EAP-TLS and EAP-PEAP). This value should be set
#   small enough to make the EAP messages fit in MTU of the network
#   interface used for EAPOL. The default value is suitable for most
#   cases.
#
# EAP-FAST variables:
# pac_file: File path for the PAC entries. wpa_supplicant will need to be
able
#   to create this file and write updates to it when PAC is being
#   provisioned or refreshed. Full path to the file should be used since
#   working directory may change when wpa_supplicant is run in the
#   background. Alternatively, a named configuration blob can be used by
#   setting this to blob://<blob name>
# phasel: fast_provisioning option can be used to enable in-line provisioning
#   of EAP-FAST credentials (PAC):
#       0 = disabled,
#       1 = allow unauthenticated provisioning,
#       2 = allow authenticated provisioning,
#       3 = allow both unauthenticated and authenticated provisioning
# fast_max_pac_list_len=<num> option can be used to set the maximum
#   number of PAC entries to store in a PAC list (default: 10)
# fast_pac_format=binary option can be used to select binary format for
#   storing PAC entries in order to save some space (the default
#   text format uses about 2.5 times the size of minimal binary
```



```
#          format)
#
# wpa_supplicant supports number of "EAP workarounds" to work around
# interoperability issues with incorrectly behaving authentication servers.
# These are enabled by default because some of the issues are present in
# large
# number of authentication servers. Strict EAP conformance mode can be
# configured by disabling workarounds with eap_workaround=0.

# Station inactivity limit
#
# If a station does not send anything in ap_max_inactivity seconds, an
# empty data frame is sent to it in order to verify whether it is
# still in range. If this frame is not ACKed, the station will be
# disassociated and then deauthenticated. This feature is used to
# clear station table of old entries when the STAs move out of the
# range.
#
# The station can associate again with the AP if it is still in range;
# this inactivity poll is just used as a nicer way of verifying
# inactivity; i.e., client will not report broken connection because
# disassociation frame is not sent immediately without first polling
# the STA with a data frame.
# default: 300 (i.e., 5 minutes)
#ap_max_inactivity=300

# DTIM period in Beacon intervals for AP mode (default: 2)
#dtim_period=2

# Beacon interval (default: 100 TU)
#beacon_int=100

# disable_ht: Whether HT (802.11n) should be disabled.
# 0 = HT enabled (if AP supports it)
# 1 = HT disabled
#
# disable_ht40: Whether HT-40 (802.11n) should be disabled.
# 0 = HT-40 enabled (if AP supports it)
# 1 = HT-40 disabled
#
# disable_sgi: Whether SGI (short guard interval) should be disabled.
# 0 = SGI enabled (if AP supports it)
# 1 = SGI disabled
#
# ht_mcs:  Configure allowed MCS rates.
#  Parsed as an array of bytes, in base-16 (ascii-hex)
# ht_mcs=""                                // Use all available (default)
# ht_mcs="0xff 00 00 00 00 00 00 00 00 00 " // Use MCS 0-7 only
# ht_mcs="0xff ff 00 00 00 00 00 00 00 00 "  // Use MCS 0-15 only
#
# disable_max_amsdu:  Whether MAX_AMSDU should be disabled.
# -1 = Do not make any changes.
# 0  = Enable MAX-AMSDU if hardware supports it.
# 1  = Disable AMSDU
#
# ampdu_density:  Allow overriding AMPDU density configuration.
#  Treated as hint by the kernel.
```



```
# -1 = Do not make any changes.
# 0-3 = Set AMPDU density (aka factor) to specified value.

# disable_vht: Whether VHT should be disabled.
# 0 = VHT enabled (if AP supports it)
# 1 = VHT disabled
#
# vht_capa: VHT capabilities to set in the override
# vht_capa_mask: mask of VHT capabilities
#
# vht_rx_mcs_nss_1/2/3/4/5/6/7/8: override the MCS set for RX NSS 1-8
# vht_tx_mcs_nss_1/2/3/4/5/6/7/8: override the MCS set for TX NSS 1-8
# 0: MCS 0-7
# 1: MCS 0-8
# 2: MCS 0-9
# 3: not supported

# Example blocks:

# Simple case: WPA-PSK, PSK as an ASCII passphrase, allow all valid ciphers
network={
    ssid="simple"
    psk="very secret passphrase"
    priority=5
}

# Same as previous, but request SSID-specific scanning (for APs that reject
# broadcast SSID)
network={
    ssid="second ssid"
    scan_ssid=1
    psk="very secret passphrase"
    priority=2
}

# Only WPA-PSK is used. Any valid cipher combination is accepted.
network={
    ssid="example"
    proto=WPA
    key_mgmt=WPA-PSK
    pairwise=CCMP TKIP
    group=CCMP TKIP WEP104 WEP40
    psk=06b4be19da289f475aa46a33cb793029d4ab3db7a23ee92382eb0106c72ac7bb
    priority=2
}

# WPA-Personal(PSK) with TKIP and enforcement for frequent PTK rekeying
network={
    ssid="example"
    proto=WPA
    key_mgmt=WPA-PSK
    pairwise=TKIP
    group=TKIP
    psk="not so secure passphrase"
    wpa_ptk_rekey=600
}
```

```
# Only WPA-EAP is used. Both CCMP and TKIP is accepted. An AP that used
WEP104
# or WEP40 as the group cipher will not be accepted.
network={
    ssid="example"
    proto=RSN
    key_mgmt=WPA-EAP
    pairwise=CCMP TKIP
    group=CCMP TKIP
    eap=TLS
    identity="user@example.com"
    ca_cert="/etc/cert/ca.pem"
    client_cert="/etc/cert/user.pem"
    private_key="/etc/cert/user.prv"
    private_key_passwd="password"
    priority=1
}

# EAP-PEAP/MSCHAPv2 configuration for RADIUS servers that use the new
peaplabel
# (e.g., Radiator)
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=PEAP
    identity="user@example.com"
    password="foobar"
    ca_cert="/etc/cert/ca.pem"
    phase1="peaplabel=1"
    phase2="auth=MSCHAPV2"
    priority=10
}

# EAP-TTLS/EAP-MD5-Challenge configuration with anonymous identity for the
# unencrypted use. Real identity is sent only within an encrypted TLS tunnel.
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=TTLS
    identity="user@example.com"
    anonymous_identity="anonymous@example.com"
    password="foobar"
    ca_cert="/etc/cert/ca.pem"
    priority=2
}

# EAP-TTLS/MSCHAPv2 configuration with anonymous identity for the unencrypted
# use. Real identity is sent only within an encrypted TLS tunnel.
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=TTLS
    identity="user@example.com"
    anonymous_identity="anonymous@example.com"
    password="foobar"
    ca_cert="/etc/cert/ca.pem"
    phase2="auth=MSCHAPV2"
```

```
}

# WPA-EAP, EAP-TTLS with different CA certificate used for outer and inner
# authentication.
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=TTLS
    # Phase1 / outer authentication
    anonymous_identity="anonymous@example.com"
    ca_cert="/etc/cert/ca.pem"
    # Phase 2 / inner authentication
    phase2="auth=TLS"
    ca_cert2="/etc/cert/ca2.pem"
    client_cert2="/etc/cer/user.pem"
    private_key2="/etc/cer/user.prv"
    private_key2_passwd="password"
    priority=2
}

# Both WPA-PSK and WPA-EAP is accepted. Only CCMP is accepted as pairwise and
# group cipher.
network={
    ssid="example"
    bssid=00:11:22:33:44:55
    proto=WPA RSN
    key_mgmt=WPA-PSK WPA-EAP
    pairwise=CCMP
    group=CCMP
    psk=06b4be19da289f475aa46a33cb793029d4ab3db7a23ee92382eb0106c72ac7bb
}

# Special characters in SSID, so use hex string. Default to WPA-PSK, WPA-EAP
# and all valid ciphers.
network={
    ssid=00010203
    psk=000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f
}

# EAP-SIM with a GSM SIM or USIM
network={
    ssid="eap-sim-test"
    key_mgmt=WPA-EAP
    eap=SIM
    pin="1234"
    pcsc=""
}

# EAP-PSK
network={
    ssid="eap-psk-test"
    key_mgmt=WPA-EAP
    eap=PSK
    anonymous_identity="eap_psk_user"
    password=06b4be19da289f475aa46a33cb793029
```

```
        identity="eap_psk_user@example.com"
    }

# IEEE 802.1X/EAPOL with dynamically generated WEP keys (i.e., no WPA) using
# EAP-TLS for authentication and key generation; require both unicast and
# broadcast WEP keys.
network={
    ssid="lx-test"
    key_mgmt=IEEE8021X
    eap=TLS
    identity="user@example.com"
    ca_cert="/etc/cert/ca.pem"
    client_cert="/etc/cert/user.pem"
    private_key="/etc/cert/user.prv"
    private_key_passwd="password"
    eapol_flags=3
}

# LEAP with dynamic WEP keys
network={
    ssid="leap-example"
    key_mgmt=IEEE8021X
    eap=LEAP
    identity="user"
    password="foobar"
}

# EAP-IKEv2 using shared secrets for both server and peer authentication
network={
    ssid="ikev2-example"
    key_mgmt=WPA-EAP
    eap=IKEV2
    identity="user"
    password="foobar"
}

# EAP-FAST with WPA (WPA or WPA2)
network={
    ssid="eap-fast-test"
    key_mgmt=WPA-EAP
    eap=FAST
    anonymous_identity="FAST-000102030405"
    identity="username"
    password="password"
    phase1="fast_provisioning=1"
    pac_file="/etc/wpa_supplicant.eap-fast-pac"
}

network={
    ssid="eap-fast-test"
    key_mgmt=WPA-EAP
    eap=FAST
    anonymous_identity="FAST-000102030405"
    identity="username"
    password="password"
}
```

```
        phase1="fast_provisioning=1"
        pac_file="blob://eap-fast-pac"
    }

# Plaintext connection (no WPA, no IEEE 802.1X)
network={
    ssid="plaintext-test"
    key_mgmt=NONE
}

# Shared WEP key connection (no WPA, no IEEE 802.1X)
network={
    ssid="static-wep-test"
    key_mgmt=NONE
    wep_key0="abcde"
    wep_key1=0102030405
    wep_key2="1234567890123"
    wep_tx_keyidx=0
    priority=5
}

# Shared WEP key connection (no WPA, no IEEE 802.1X) using Shared Key
# IEEE 802.11 authentication
network={
    ssid="static-wep-test2"
    key_mgmt=NONE
    wep_key0="abcde"
    wep_key1=0102030405
    wep_key2="1234567890123"
    wep_tx_keyidx=0
    priority=5
    auth_alg=SHARED
}

# IBSS/ad-hoc network with WPA-None/TKIP.
network={
    ssid="test adhoc"
    mode=1
    frequency=2412
    proto=WPA
    key_mgmt=WPA-NONE
    pairwise=NONE
    group=TKIP
    psk="secret passphrase"
}

# Catch all example that allows more or less all configuration modes
network={
    ssid="example"
    scan_ssid=1
    key_mgmt=WPA-EAP WPA-PSK IEEE8021X NONE
    pairwise=CCMP TKIP
    group=CCMP TKIP WEP104 WEP40
}
```

```
    psk="very secret passphrase"
    eap=TTLS PEAP TLS
    identity="user@example.com"
    password="foobar"
    ca_cert="/etc/cert/ca.pem"
    client_cert="/etc/cert/user.pem"
    private_key="/etc/cert/user.prv"
    private_key_passwd="password"
    phasel="peaplabel=0"
}

# Example of EAP-TLS with smartcard (openssl engine)
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=TLS
    proto=RSN
    pairwise=CCMP TKIP
    group=CCMP TKIP
    identity="user@example.com"
    ca_cert="/etc/cert/ca.pem"
    client_cert="/etc/cert/user.pem"

    engine=1

    # The engine configured here must be available. Look at
    # OpenSSL engine support in the global section.
    # The key available through the engine must be the private key
    # matching the client certificate configured above.

    # use the opensc engine
    #engine_id="opensc"
    #key_id="45"

    # use the pkcs11 engine
    engine_id="pkcs11"
    key_id="id_45"

    # Optional PIN configuration; this can be left out and PIN will be
    # asked through the control interface
    pin="1234"
}

# Example configuration showing how to use an inlined blob as a CA
certificate
# data instead of using external file
network={
    ssid="example"
    key_mgmt=WPA-EAP
    eap=TTLS
    identity="user@example.com"
    anonymous_identity="anonymous@example.com"
    password="foobar"
    ca_cert="blob://exampleblob"
    priority=20
}
```

```
blob-base64-exampleblob={
SGVsbG8gV29ybGQhCg==
}
```

```
# Wildcard match for SSID (plaintext APs only). This example select any
# open AP regardless of its SSID.
```

```
network={
    key_mgmt=NONE
}
```

```
# Example config file that will only scan on channel 36.
```

```
freq_list=5180
```

```
network={
    key_mgmt=NONE
```