

Hawkeye

Release 6.0.7

XRPi User Guide

Notices

Copyright Notice

© Keysight Technologies 2016–2025

No part of this document may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Keysight Technologies, Inc. as governed by United States and international copyright laws.

Warranty

The material contained in this document is provided “as is,” and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Keysight disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Keysight shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Keysight and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

U.S. Government Rights

The Software is “commercial computer software,” as defined by Federal Acquisition Regulation (“FAR”) 2.101. Pursuant to FAR 12.212 and 27.405-3 and Department of Defense FAR Supplement (“DFARS”) 227.7202, the U.S. government acquires commercial computer software under the same terms by which the software is customarily provided to the public. Accordingly,

Keysight provides the Software to U.S. government customers under its standard commercial license, which is embodied in its End User License Agreement (EULA), a copy of which can be found at <http://www.keysight.com/find/sweula>. The license set forth in the EULA represents the exclusive authority by which the U.S. government may use, modify, distribute, or disclose the Software. The EULA and the license set forth therein, does not require or permit, among other things, that Keysight: (1) Furnish technical information related to commercial computer software or commercial computer software documentation that is not customarily provided to the public; or (2) Relinquish to, or otherwise provide, the government rights in excess of these rights customarily provided to the public to use, modify, reproduce, release, perform, display, or disclose commercial computer software or commercial computer software documentation. No additional government requirements beyond those set forth in the EULA shall apply, except to the extent that those terms, rights, or licenses are explicitly required from all providers of commercial computer software pursuant to the FAR and the DFARS and are set forth specifically in writing elsewhere in the EULA. Keysight shall be under no obligation to update, revise or otherwise modify the Software. With respect to any technical data as defined by FAR 2.101, pursuant to FAR 12.211 and 27.404.2 and DFARS 227.7102, the U.S. government acquires no greater than Limited Rights as defined in FAR 27.401 or DFAR 227.7103-5 (c), as applicable in any technical data. 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Contacting us

Keysight headquarters

1400 Fountaingrove Parkway
 Santa Rosa, CA 95403-1738
 Email address: support@keysight.com

Support

Location	Phone number	Local time
<i>Americas</i>		
US, Canada	1-888-829-5558	8h00 – 17h00
Brazil	0800-892-0522	8h00 – 17h00
Mexico	001-888-829-5558	8h00 – 17h00
Other	+1-719-273-6516	8h00 – 17h00
<i>EMEA</i>		
Belgium	0800-18686	8h30 – 17h30
Finland	0800-913-352	8h30 – 17h30
France	0800-917228	8h30 – 17h30
Germany	0800-0824099	8h30 – 17h30
India	1800-18-02552	8h30 – 17h30
Ireland	1800-949245	8h30 – 17h30
Israel	1-809-454975	8h30 – 17h30
Italy	0800-790571	8h30 – 17h30
Luxembourg	0800-25112	8h30 – 17h30
Netherlands	0800-022-9086	8h30 – 17h30

Romania	0213 015 699	8h30 – 17h30
Spain	800-654386	8h30 – 17h30
Sweden	0201-202266	8h30 – 17h30
United Kingdom	0800-0293882	8h30 – 17h30
<i>Asia and Australia</i>		
Australia	1-800-370-558	8h30 – 17h00
China Mainland	800-810-0005	8h30 – 17h30
	400-810-0005	8h30 – 17h30
Hong Kong	800-931-613	9h00 – 18h00
Japan	0120-421-621	9h00 – 17h30
Malaysia	1800-819 092	8h30 – 17h30
South Korea	080-770-0800	8h30 – 17h30
Singapore	800-101-3797	8h30 – 17h30
Taiwan	0800-699-880	9h00 – 18h00
Other	+65 6215 7600	8h30 – 17h30 (Singapore)

Table of Contents

Contacting us **3**

Chapter 1 Configure endpoints **1**

 Hawkeye endpoints definition 2

 XRPi 2

 XRPi hardware 2

 Configure XRPi 5

Index **18**

Configure endpoints

This chapter describes how to configure Hawkeye Endpoints.

In this chapter:

Hawkeye endpoints definition	2
XRPi	2
XRPi hardware	2
Configure XRPi	5

Hawkeye endpoints definition

An endpoint is called a probe and represents a device inserted at a key juncture in a network for monitoring or collecting performance data about network activity. The test traffic generated conforms to applicable standards. For example, voice traffic contains generated data that conforms to selected codecs with correct packet sizes and packet timings. The Hawkeye endpoint for corporate security generates its test data. Hawkeye does not monitor or analyze corporate private transmitted data. The following Hawkeye endpoints are available:

- Keysight Technologies hardware endpoints: XR3000, XR-E1S, IxProbe, XRPi, and xr2000
- XR software endpoints: XR-VM, XR Docker, PI4 Image, and XR Cloud
- Software endpoints: IOS EP, Android EP, Linux EP, and Windows EP

Hawkeye supports different software endpoints for various platforms. You can download these endpoints from the Keysight Technologies, Inc. web site mentioned in the section on software endpoints.

NOTE

The Real Services testing is only available with XR2000, XR2000 VM (Linux Software), XR Docker, and XRPi endpoints. Use XRPi to perform or monitor wireless network testing.

XRPi

The XRPi is a low-profile ultra-small form factor unit. It is ideal for use as an endpoint. It contains an ultra-low voltage CPU and an integrated Ethernet port. It does not contain any fans.

The optional wireless network device supports both 2.4 and 5 MHz. The wireless network device also supports the latest state 802.1ac with 80 MHz.



XRPi hardware

Package contents

XRPi ships with the following components:

- XRPi unit
- Dual-voltage (110/220) AC power adapter
- Keysight wireless network device

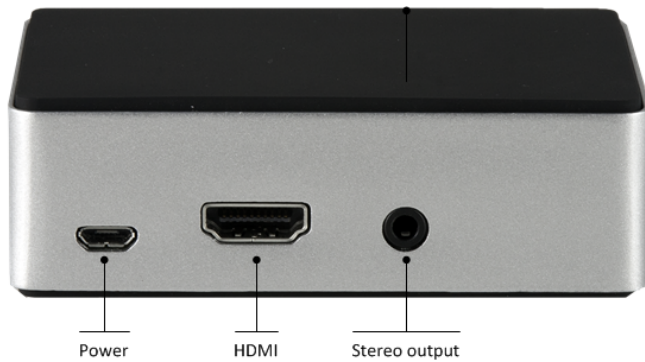
- Non-powered 2.0 USB hub
- HDMI-to-DVI cable
- Quick start guide

Ethernet and USB ports



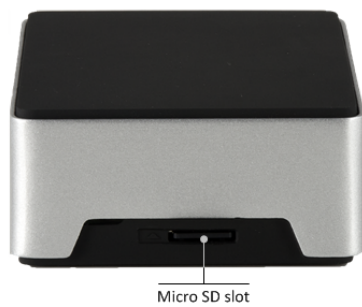
Component	Description
Ethernet port	Use these ports to connect the XR Pi to other Ethernet devices.
	LAN LEDs
	Green: Connection speed
	On 100 Mbps
	Off 10 Mbps
	Yellow: Activity
	On Port is active
USB ports	Blinking Port is sending or receiving traffic
	USB 2.0 ports.
	Keyboards, mice, USB 2.0 hub, and the wireless network device are the only supported accessories that can be plugged into these ports.

Power and HDMI ports



Component	Description
Power port	Micro-USB power port. Connect the supplied AC power adapter to this port.
HDMI	HDMI 1.4 video output port. Connect the supplied HDMI cable to this port.
Stereo	3.5 mm stereo output port (disabled).

Micro SD slot



Component	Description
Micro SD slot	Micro SD card slot. A micro SD card contains the Hawkeye operating system and application. The micro SD card is the only permanent storage on Hawkeye.

Insert the micro SD card

To insert the micro SD card, position the micro SD card with the contacts facing upwards and insert the card into the slot until you hear a click.

Remove the micro SD card

To remove the micro SD card, press the card inwards with a fingernail until you hear a click, and then withdraw the card from the slot.



System information

Current Draw	~ 650 mA, (3.0 W)
Temperature (ambient)	Operating: 0°C ~ 70°C Storage: -20°C ~ 70°C
Humidity (RH), ambient	5 ~ 95 percent, non condensing
Dimensions (WxHxD)	92 mm x 29 mm x 67 mm 3.63" x 1.13" x 2.63"
Weight	100 g
Approvals & Compliance	CE Emission, FCC Class A, RoHS

Configure XRpi

You can connect the device through HDMI and keyboard by using a USB port.



Sign in the XRpi by using the following default credentials:

- Username: root
- Password: Ixia!123

Default configuration (as shipped from factory)

Physical port	Logical port	Address
LAN1	eth0	DHCP

Configure port

This section explains the steps to configure DHCP IP and static IP.

IP configuration for eth0

The default configuration for eth0 is IPv4 DHCP to configure IPv6 or Static IPv4 Please follow the instructions below.

IPv4 using static IP only

In file /etc/network/interfaces add

```
iface eth0 inet static
address 192.168.x.x
gateway 192.168.1.1
netmask 255.255.255.1
network 192.168.1.0
broadcast 192.168.1.255
```

IPv6 using DHCP and disable IPv4 on eth0.

In file /etc/network/interfaces add

```
iface eth0 inet6 dhcp
```

and remove

```
iface eth0 inet dhcp
```

IPv4 using static IP and IPv6 using DHCP on eth0.

In file /etc/network/interfaces add

```
iface eth0 inet6 dhcp
iface eth0 inet static
address 192.168.x.x
gateway 192.168.1.1
netmask 255.255.255.1
network 192.168.1.0
broadcast 192.168.1.255
```

IPv4 using DHCP and IPv6 using static IP on eth0.

In file /etc/network/interfaces add

```
iface eth0 inet dhcp
iface eth0 inet6 static
address 2001:1800:1234:22::3fd/64
gateway 2001:1800:1234:22::1
```

Configure a DNS server

In file /etc/resolv.conf add

nameserver 8.8.8.8	# Google's public DNS server
nameserver 2606:4700:4700::1001	# public IPv6 DNS server

Reboot the XRPi for changes to take effect.

```
# reboot
```

Check the IP address locally and testing connectivity (DHCP or static)

1. Access the XRPI's console and use the following command to check the XRPI's IP address:
`ifconfig eth0`
2. Run the following command to test connectivity:
`ip route`
Look for: `default via [gateway IP]`. Ensure that `ping [gateway IP]` succeeds.
3. Run the following command to test that the XRPI can reach the internet and that DNS resolution works:

```
ping www.ixiacom.com
```

The ping should succeed and not time out.

If either the connectivity or name resolution fails, check the static IP information that you entered and confirm that the DHCP server is available.

NOTE

For XRPI wireless device, when setting up static IP addresses and connecting both wlan0 and eth0 interface, XRPI automatic routing scripts that ensure proper communications with the server will not turn on. We recommend to use DHCP configuration for this scenario.

4. The default host-name for the XRPI is `xrpi2- <last 5 digit of MAC address>`. The MAC address is on the label at the bottom of the XRPI.

XRPI Upgrade

Register with Hawkeye server

We recommend you to configure XRPI with the Hawkeye server as automatic endpoint.

If XRPI is registered to the Hawkeye server, you can perform the upgrades automatically. If XRPI is registered to the Hawkeye Server 1.0 or registered as a manual endpoint, you must upgrade the endpoint by using `probeConfigure`. We recommend you to use this upgrade strategy for Hawkeye users.

To update automatically, do the following:

1. Connect to the XRPI with SSH.
2. Type the following command to start the script: `probeConfigure`.
3. When the script prompts you, enter the following:
 - A name for the probe. You can use letters, numbers and hyphens (_) in the name. This option will set the host name and the name as displayed into Hawkeye. You can edit the name on the Hawkeye GUI.
 - The host name or IP address of the Hawkeye server (public IP, if in cloud).

NOTE

The name cannot contain spaces.

4. When the script ends, XRPI will automatically try to register.

Upgrade XRPI without Hawkeye Server

To upgrade XRPI from internet, do one of the following steps:

- When the XRPI has internet connectivity, connect to XRPI with SSH and run the following commands:

```
cd /tmp
```

```
wget https://ixiapublic.s3.amazonaws.com/hawkeye/xrpi_upgrade.tar.gz
tar zxvf xrpi_upgrade.tar.gz
cd ix_upgrade
./ixia_chariotprobe_install.sh
cd ..
rm -rf ix_upgrade
rm -rf xrpi_upgrade.tar.gz
```

- If the XRPI does not have internet connectivity, copy the file at the https://ixiapublic.s3.amazonaws.com/hawkeye/xrpi_upgrade.tar.gz location. Then FTP (or equivalent, such as scp) the file to the XRPI in the /tmp folder. Then run the procedure:

```
tar zxvf xrpi_upgrade.tar.gz
cd ix_upgrade
./ixia_chariotprobe_install.sh
cd ..
rm -rf ix_upgrade
rm -rf xrpi_upgrade.tar.gz
```

NOTE

A reboot after upgrade is necessary.

Secure XRPI2 ports

Follow the instructions in this section to secure and configure XRPI2 ports.

Ports opening requirements for Ixia endpoints

The ports that you must open for INBAND are:

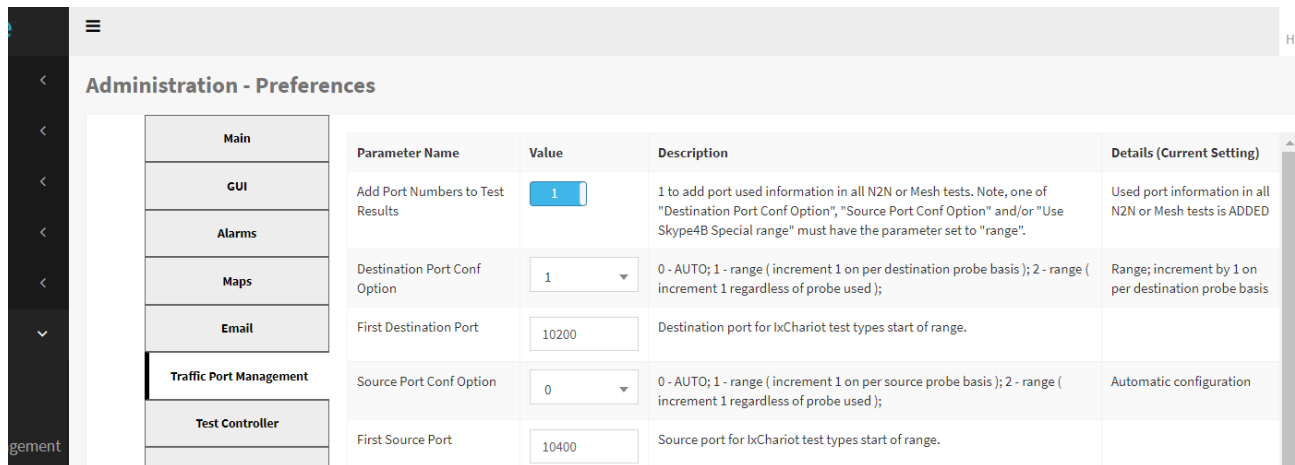
- TCP port 22: This is required for manual remote management. Hawkeye server does not need these ports. You can protect this port by access list.
- TCP/UDP ports 10115-10116. This is required for control between endpoints.
- TCP/UDP ports TEST RANGE: You must define this range in Hawkeye server (see preferences below).

We recommend to let any OUTBAND ports be opened for Real Service tests.

Configure endpoints to use specific range of ports to test traffic

On the Hawkeye Server, under the **System Preferences > Traffic Port Management**, you can configure the ports for Node to Node traffic tests.

Each test executed on the endpoint will increment this range. Hence, it is necessary to open a range on the XRPI that can allow up to the maximum number of simultaneous tests that can go to a single endpoint. For XRPI, a recyclable range of 200 ports is enough.



Configure XRPI endpoints firewall

To activate firewall on XRPI, do the following:

1. SSH as root on the XRPI device.
2. Create and edit new file with vi or nano text editor.
nano /etc/init.d/firewall
3. Add and adapt the following in the table for ports openings:

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: firewall rules
# Required-Start: $remote_fs $syslog
# Required-Stop: $remote_fs $syslog
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start daemon at boot time
# Description: Enable service provided by daemon.
### END INIT INFO
# Remove rules
iptables -F
iptables -X
# Block everything by default
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
# Keep established connections
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
# Authorize loopback (127.0.0.1)
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
# SSH (22)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
# endpoint management (10115:10116)
iptables -A INPUT -p tcp --dport 10115:10116 -j ACCEPT
```

```

iptables -A INPUT -p udp --dport 10115:10116 -j ACCEPT
# endpoint traffic (need to be adapted to console configuration)
iptables -A INPUT -p tcp --dport 50000:50500 -j ACCEPT
iptables -A INPUT -p udp --dport 50000:50500 -j ACCEPT
# ICMP (Ping)
iptables -A INPUT -p icmp -j ACCEPT
# Limit denial or service
iptables -A FORWARD -p tcp --syn -m limit --limit 1/second -j ACCEPT
iptables -A FORWARD -p udp -m limit --limit 1/second -j ACCEPT
iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/second -j ACCEPT
# protects port scans
iptables -A FORWARD -p tcp --tcp-flags SYN,ACK,FIN,RST RST -m limit --limit 1/s -j
ACCEPT
# Drop XMAS / NULL scans
iptables -A INPUT -p tcp --tcp-flags FIN,URG,PSH FIN,URG,PSH -j DROP
iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP
iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
iptables -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j DROP

```

4. Save the file and make it executable.

```
# chmod +x /etc/init.d/firewall
```

5. Start the firewall.

```
# ./etc/init.d/firewall
```

We recommend you to make firewall load automatically at boot.

```
# update-rc.d firewall defaults
```

Enable wireless network interface

You can enable the wireless network interface on XRPi2. Before you enable the wireless network interface on the XRPi2, ensure that you have the following:

- Wireless dongle provided by Keysight.
- Network connection for XRPi eth0 (LAN) interface.
- The XRPi has image with version 2.0.9062115 or later. To see the XRPi version, use the `xrpi2_` `version` command after you have SSH or use **Probe Management**.
- For 2.2 onwards releases, XRPi supports static IP address being used for eth0 but DHCP must be used for wireless network interface (WLAN0). You must configuring the static IP address for eth0, which is used for Hawkeye management. The automatic wireless network scripts configures the WLAN0 interface routes.
 - Default routes for traffic will be over wireless interface (wlan0), if it is turned on. If, however, eth0 and wlan0 interface are on different subnetworks for Mesh tests, turn on the **Set test IP that matches default gateway** parameter on the **Administration > Preferences > Advanced Options** tab, so that for mesh tests, traffic will be made on the interface that has default gateway defined. When wlan0 is defined, default gateway is configured in the routing tables to be on wlan0.
 - We recommended you not to configure the XRPi with just the wireless network interface, as eth0 is used for Hawkeye management and it is a more reliable interface.
 - The wlan0 subnet must be different to eth0 subnet for XRPi routing scripts to correctly set up routes for DHCP.

To turn on the wireless network interface on XRPi, do the following:

1. Connect the network cable to the XRPi ethernet port.
2. Connect AC power supply using micro USB power connector.
3. Use a software tool like Putty or Bitwise to connect using SSH to the XRPi eth0 interface.

NOTE

The default username is **root** and password is **Ixia!123**.

4. Connect the Ixia wireless device dongle to any available USB port on XRPi device. The wireless interface (wlan0) is automatically configured on XRPi when you connect the dongle.

```
Linux xrp-08914 4.1.20-v7+ #40 SMP Mon May 9 15:57:26 EDT 2016 armv7l
IXIA Raspberry PI image
This image is provided by Ixia
The exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Last login: Mon Oct 3 20:30:51 2016 from 192.168.83.138
root@xrp-08914:~# xrp2_version
2.0.9129825
root@xrp-08914:~# ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:20:89:14
          inet addr:192.168.82.22  Bcast:192.168.82.255  Mask:255.255.255.0
          inet6 addr: fe80::ba27:ebff:fe20:8914/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:244 errors:0 dropped:0 overruns:0 frame:0
          TX packets:203 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:40377 (39.4 KiB)  TX bytes:30216 (29.5 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0     Link encap:Ethernet  HWaddr 74:da:38:73:4f:39
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@xrp-08914:~#
```

5. Run the Linux **ifconfig** command (see the below example) to confirm the interface wlan0 is present.

NOTE

If you want to use wireless network for **Wi-Fi Inspect** and **Wi-Fi connect**, you can skip the steps above.

Route for Eth0 and Wlan0

Reboot

When eth0 (LAN) and wlan0 (Wi-Fi) are both used, XRPI will have sequential configuration for network stacks that are described below:

- The eth0 interface is brought up by system and runs dhcp query to obtains def GW, NTP server, DHCP lease, NIS servers, DNS servers, DHCP IP on LAN interface.
- A static route is configured to Hawkeye server through LAN interface to make sure the connectivity to Hawkeye server will remain through LAN interface.
- wlan0 interface is brought up by the system to connect to the access point. DHCP runs first, then the custom XRPI scripts run. It reconfigures and overwrites the routing tables, specifically the global default route (default GW for DHCP) to switch to wlan0. Then, the default routes and network routes are set in the global tables and the interface table. If eth0 and wlan0 are on different subnets, all routes to the subnet are unmodified. If they are on the same subnet, then this may cause routing issues as this is not a supported configuration.

Node to Node tests

Interface to be used for traffic is specified as part of test creation on Hawkeye server.

Mesh and real services tests

- If eth0 and wlan0 are on the same subnet, this may cause routing issues. This is not a supported configuration.
- If eth0 and wlan0 are on different subnets, (network routes apply) so if destination IP for test is on subnet for eth0, traffic will go on eth0.
- If destination IP for test is on the subnet for wlan0, traffic will go on wlan0.

Wireless network connectivity lost

- If XRPI receives disconnect from AP or a DHCP release, XRPI will attempt to reconnect by using wpa_default.cfg, as long as DHCP lease has not expired.
- If XRPI receives no message/event for lost connectivity (eg AP loses power), XRPI will attempt to reconnect every 10 seconds.

NOTE

Rebooting triggers re-establishment of wlan0.

Connect to wireless network

You can connect to wireless network using one of the following:

- By using **Wi-Fi Connect** of Hawkeye Server GUI.
- By using Real Service wireless network connect test.
- By using a script (wifiConnect) to connect to specific SSID.

NOTE

The double quote symbol is not supported in the password. It is advised that if manually editing the wpa_default.cfg file not to use the double quote symbol for comments as it is used by the parser to denote parameters.

Use Hawkeye server GUI option for Wi-Fi Connect

The Hawkeye server supports the ability to configure the wireless network interface for XRPi, assuming eth0 is present. After XRPi is configured as an automatic endpoint on the Hawkeye server in the **Probe Management > Probe Management Automatic**, if the you drill down on a specific XRPi, there is a pop-up for **Wi-Fi Configure**, which allows you to enter details for the AP. When **connect** is selected, XRPi uses these details to connect to the AP. If successful, it saves the changes on XRPi in the **wpa_default.cfg** file.

Use Real Service Wi-Fi Connect test

The Real Service Wi-Fi Connect test is explained in the *XR2000 User Guide > Real services* section for the Wi-Fi Connect test type. When the test is run, the XRPi receives commands to directly configure the wireless network device. XRPi releases wlan0 connection (routes updated), then attempts to connect to the AP defined for the Wi-Fi Connect test. The connection details from the test are temporary and are not saved in a file on the XRPi. If the connection is successful, routes are updated for wlan0 and results are sent to Hawkeye Server. After the test is finished, if Wi-Fi Connect test parameter **reset Interface** is set to **yes**, wlan0 is released and wireless network reestablishment occurs to the AP defined in wpa_default.cfg (if file exists). If Wi-Fi Connect test parameter **reset Interface** is set to **no**, wlan0 will remain with temporary test configuration until XRPi is restarted. The ethernet port must be active for the management connection to the Hawkeye Server to report results, as the test resets the wlan0 connection.

Use script to connect to specific SSID

The XRPi has a custom wireless network connection script that can be used to connect to an access point (SSID). The script is located in /home/ixia directory and is named wifiConnect.py:

```
# cd /home/ixia
# wifiConnect
```

The script scans for available AP's in range, then prompt for the selection of a SSID/BSSID. You can then enter authentication parameters. If the wifiConnect script runs with the **-D** option to save the connection as default, it saves in the directory /etc/wpa_supplicant as wpa_default.cfg if only the connection is successful. If the **-D** option is not used with the wifiConnect script command, it prompts if you want to save settings as default. If an AP has a configuration that is not addressed by the wifiConnect script, you have to manually edit the parameters in the wpa_default.cfg file.

There are several options that can be used with the wifiConnect command:

- To list out all options with a brief description:

```
# wifiConnect -h
```

- To connect then save the AP connection details as default:

```
# wifiConnect -D
```

- To use a pre-existing AP configuration file:

```
# wifiConnect --config /etc/wpa_supplicant/wifi_ixia_guest.cfg
```

Disable wireless interface

To disable wireless interface, do the following:

1. Go to XRPI (ssh) and physically renaming the XRPI WIFI configuration file (/etc/wpa_supplicant/wpa_default.cfg) to another name or delete the file.
2. Restart XRPI or bring down the wireless interface with the command:

```
Ifdown wlan0
```

Configure custom wireless network

Hawkeye Server supports the use of wireless network custom parameters (defined by `wpa_supplicant`). , The specific parameter, however, must be supported by the `wpa_supplicant` and the physical wireless network device.

When you perform a wireless network connect to an AP, following steps occur:

1. The configuration including custom wireless network configuration is saved into the `wpa_default.cfg` file on the XRPI.
2. The `wpa_supplicant`, which is an IEEE 802.1 cross platform compliant application (third party Linux app), reads the configuration from `wpa_default.cfg` and sends information to the wireless network driver specific to the wireless network device. After you configure the wireless network device, the `wpa_supplicant` negotiates connection with the AP and radius server for authentication.

IMPORTANT

Hawkeye can only take custom wireless network parameters and store them in the `wpa_default.cfg` file to be read by the Linux `wpa_supplicant` application. For your custom wireless network parameters to be successful, the version of `wpa_supplicant` must support the parameters and the wireless network driver must also support the custom parameter. Refer online to all parameters supported by the `wpa_supplicant`. Contact Keysight Customer support if you need confirmation that a specific parameter is implemented or supported. We recommend to use the minimum amount of configuration in the `wpa_default` file (eg. key management, user, password, SSID).

Troubleshoot wireless network issues

This section provides you information with scenarios to troubleshoot the wireless network issues.

1. Confirm hardware and drivers are okay for wireless network:

- a. Confirm wireless network device is plugged into XRPi USB port and has been initialized by checking the LED on the **Edimax** labeled wireless network device. A blue flash from the LED indicates the device is working.
 - b. Run the following Linux command to confirm wireless network device detected in USB port. After detecting the device, the name of the device that is **Edimax** appears in the list of devices.


```
# lsusb -t | grep Edimax
```
 - c. Confirm that the driver for **Edimax** device is running, using the following command:


```
# lsmod | grep 8812au
```
- Confirm WLAN0 interface for wireless network is operational:
 - a. To confirm if there is an entry and the wlan0 interface is operational, use the following command:


```
# "ifconfig wlan0"
```
 - b. To check the status of WLAN, use the following command:


```
# ip link show wlan0
```
 - c. To bring up the Wi-Fi interface, use the following command:


```
# ip link set wlan0 up
```
 - d. To check the log files for time stamp for wireless network connectivity and release events, use the following command:


```
# cat /var/log/Ixia
```
 - Check if the wireless network interface is connected to an AP by using the following command:


```
# iw wlan0 link
```
 - Confirm if the XRPi Wi-Fi can see all AP in range using the following command:


```
# iw wlan0 scan
```
 - Confirm the IP assigned to WLAN0, using the following command:


```
# ifconfig wlan0
```
 - After an IP is assigned to wlan0, confirm all routes including the default route is using wlan0 with the following command:


```
# route -n
```
 - If an XRPi wireless network endpoint is moved from one location to another, the IP addresses for wlan0 and eth0 changes. After the XRPi Wi-Fi endpoint is located in its new location, use the following commands in the given sequence to optimize the routes:


```
# route -n
# ip route flush table main
# rm -rf /etc/wpa_supplicant/wpa_default.cfg
# wlan0 down
# wlan0 up
# route -n
# ifconfig
```
 - Schedule the real service test **Wi-Fi Inspect** on the Hawkeye server at an interval of five or ten minutes for one or two hours to find problems with different AP. View results in the **Wi-Fi**

Dashboard. The results show the stability of each AP signal strength over the period. Graphical representation of the results, channels, and signal strength of each selected AP issues of all AP is available, which helps you to view the overlapping interference:

- Different wireless network devices report signal strength at different layers in the wireless network stack, so sometimes there can be differences when comparing.
- The Edimax device considers a weak signal to be below -65 dBm. Packets can become corrupted and lost at this weak signal strength. The Edimax device may have difficulty connecting to an AP with a very weak signal strength of -70 dB.
- The wireless network scan may have issues detecting and reporting older AP if they do not provide primary channel information and/or frequency.

9. If connection to an AP is unsuccessful, perform the following steps to troubleshoot:

- a. Perform a Wi-FiConnect test from the XRPI CLI and enable debug logs using one of the following commands, to see message sequences and reason codes.

```
# wifiConnect --debug=6
# wifiConnect --debug=6 2>&1 |tee /tmp/myLog.txt
```

NOTE

Set debug flag to **7** for most detailed logs, which include messaging and state changes. Second command saves trace to a file.

- b. Perform a detailed wireless network scan to find other AP using the same channel as the one previously attempting to connect to. The noise interference for an AP is affected by the relative signal strength of other AP's using the same channel and the signal strength of those AP. The bandwidth used by an AP usually spans the channel number above and below the reported channel number. The following command will perform a detailed wireless network scan and save results to a file:

```
# /home/ixia/wifiScan.py -d 4 -u 2>&1 |tee /tmp/myScan.log
```

10. Fails to connect or drop from specific BSSID. If you specify BSSID with wifiConnect command or test the connection may drop or lose. A disconnect from an AP (BSSID) may be due to a local generation event or remote generation event. A locally generated event for disconnect can be due to XRPI wireless network inability to see the beacon or keep it alive from its connected AP (BSSID). Keysight recommends you to use different signal channels for each AP (BSSID) of an SSID with multiple AP. It is possible that an AP for the same SSID, but different BSSID, may have a stronger signal and drown out the beacon of the weaker BSSID when using the same signal channel.

The connection can also drop due to the remote generation of events. The Radius server may realize that the XRPI Wi-Fi is connected to an AP (BSSID) but an AP (BSSID) belonging to the same SSID (i.e. Corporate Wi-Fi) may have a stronger signal. The Radius server will often inform the currently connected AP to send a disconnect and optionally perform a handover to a different stronger signal strength AP (BSSID).

11. You can manually adjust routes. The following code snippet is for reference. We suggest you to first check if the default route is missing.

```
# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 0.0.0.0 0.0.0.0 U 0 0 0 wlan0
172.16.0.76 0.0.0.0 255.255.255.255 UH 0 0 0 eth0
```

The above output shows the missing route.

```
# ifdown wlan0; ifup wlan0
```

Then, verify the routes with

```
# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 0.0.0.0 0.0.0.0 U 0 0 0 wlan0
172.16.0.0 0.0.0.0 255.255.255.128 U 0 0 0 wlan0
172.16.0.76 0.0.0.0 255.255.255.255 UH 0 0 0 eth0
```

Finally, confirm the additional route for wlan0 is present and routes are as expected.

NOTE

The microSD card cannot be moved from one XRPi to another as the identity from the first XRPi will be used for the second XRPi.

Index

	C
customer assistance	3
	P
product support	3
	T
technical support	3
	X
XRPi content, XRPi system information	2
XRPi setup, configure XRPi	5



© Keysight Technologies, 2025

This information is subject to change without notice.

www.keysight.com

913-2383-01 Rev A