Wind River

User's Guide 3rd Ed

Last Updated: October 15, 2015

Table of Contents

Chapter 1

Introduction and Overview

1.1 Wind River Intelligent Device Platform Overview

The Wind River Intelligent Device Platform XT (IDP XT) packages a commercial-grade Wind River Linux development platform with security and management tools for gateways.

IDP XT provides integrated development and management support for distributed systems that utilize smart services with cloud computing. It includes secure remote management layer for cloud-based smart services, including automated customer interaction and support.

Included in IDP XT

- Wind River Linux
- Wind River Workbench
- Wind River Intelligent Device Platform XT
- McAfee Embedded Control

This guide describes how to set up and run the AAEON AIOT Quark SoC X1000 Kit.

1.2 Included in Yocto

The Yocto Project accomplishes the following:

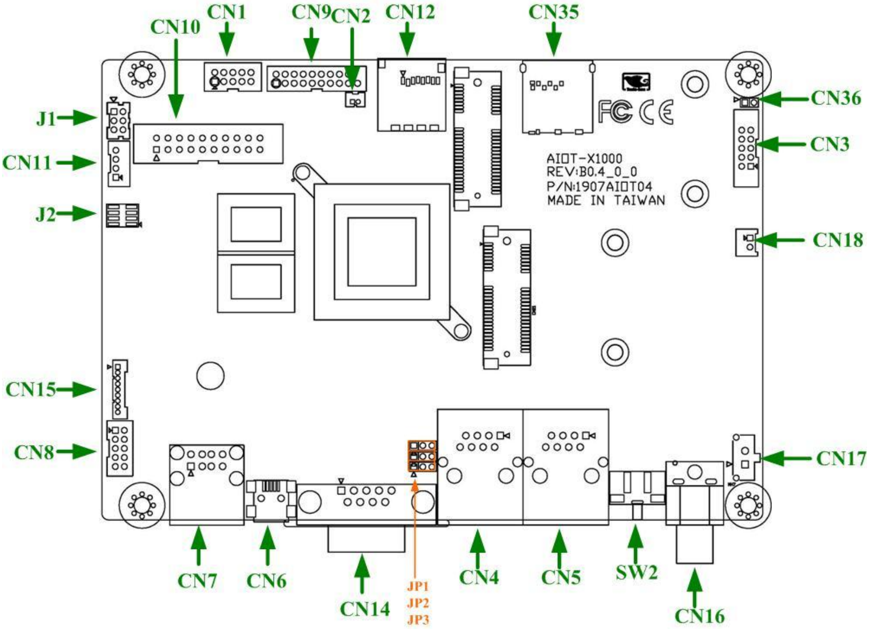
- Co-maintains and leverages Bitbake and OpenEmbedded-Core, and extends them by adding COTS BSPs, a reference distribution, documentation, etc.
- Provides a tested, pre-prepared combination of build system components
- Includes autobuilder sessions
- QA testings
- Eclipse Plugins
- Branding / Compatibility Program
- …etc…

This guide describes how to set up and run the AAEON AIOT Quark SoC X1000 Kit.

Chapter 2

Platform Setup

2.1 Board Layout



2.2 List of Connectors

Label	Function	Connector Type
CN1	JTAG Programming Port	(TF)BOX HEADER.5*2P.180D(M).DIP.2.0mm
CN2	Batter	(TF)WAFER BOX.2P.180D.(M).1.25mm
CN3	ADC	(TF)BOX HEADER.5*2P.180D(M).DIP.2.0mm.
CN4	10/100 RJ45	(TF)RJ45.12P.90D(F).W/Transformer & LED.DIP
CN5	10/100 RJ45	(TF)RJ45.12P.90D(F).W/Transformer & LED.DIP
CN6	MINI USB	(TF)MINI USB CONNECTOR R/A 0.8.R/A 0.8mm.5P.90D(F)
CN7	DUAL USB	(TF)USB CONNECTOR DUAL PORT.8P.90D.(F).for USB2.0
CN8	DUAL USB	(TF)BOX HEADER.5*2P.180D.(M).2.00mm.Narrow Frame.DIP
CN9	GPIO	(TF)BOX HEADER.10*2P.180D(M).DIP.2.0mm.Narrow Frame
CN10	ZIGBEE / ENERGY SPI or UART MODULE	(TF)BOX HEADER.10*2P.180D.(M).2.54mm.
CN11	I2C	(TF)WAFER BOX.4P.180D.(M).2.0mm.W/LOCK DIP

CN12	Micro-SD Card	(AOH)(TF)Micro SD SKT.8P.90D(F).SMD.Push-Push type
CN14	Serial Port RS232/RS485/RS422	(TF)D-SUB CONNECTOR.9P.90D(M).DIP.Green.
CN15	Serial Port RS232/RS485/RS422	(TF)WAFER BOX.9P.180D(M).DIP.1.25mm.
CN16	DC Input	(TF)DC Power Jack.3P.90D(F).
CN17	DC Input	(TF)WAFER BOX.2*1P.180D(M).DIP.3.0mm.
CN18	Power LED	(TF)WAFER BOX.2P.180D.(M).2.0mm.W/LOCK DIP.
CN36	Micro-SD LED	(TF)PIN HEADER.2*1P.180D.(M).2.0mm.DIP
CN20	Full Mini PCIE	(TF)MiniCard SLOT.52P.90D.(F).SMD
CN21	Half Mini PCIE	(TF)MiniCard SLOT.52P.90D.(F).SMD
J1	RESET	(TF)WAFER BOX.6P.180D(M).2.0mm.W/LOCK DIP.
J2	SPI Flash	(TF)PIN HEADER.4*2P.180D.(M).1.27mm.SMD.W/Cap.

2.3 Connecting to Target System (Board)

The platform is designed as a headless device and does not support KVM (Keyboard, Video, Mouse). You must connect remotely via one of the following methods:

- Terminal emulation over a serial connection (RS-232 or RS-485). See Section 2.3.1
- SSH over a wired network connection. See Section 2.3.2
- SSH over a wireless network connection. See Section 2.3.3

2.3.1 Serial Connection

To update the firmware and install IDP runtime on the target (board), it is necessary to connect the target (board) with a terminal emulator using the provided serial cable. The example below assumes you are using Putty.

1. Connect the target (board) to the host computer via the RS-232 debug console port, using the provided 3.5 mm to DB-9 cable and optional DB-9 to USB adapter.
2. Turn on the platform. A device is created: /dev/ttyS0
3. Run the terminal emulator on the host computer using one of the following commands:

```
# sudo putty &
```

or

```
# gksudo putty to run Putty as root
```

or

```
# sudo chmod 666 /dev/ttyUSB0
```

Use the following settings:

a. Speed = 115,200

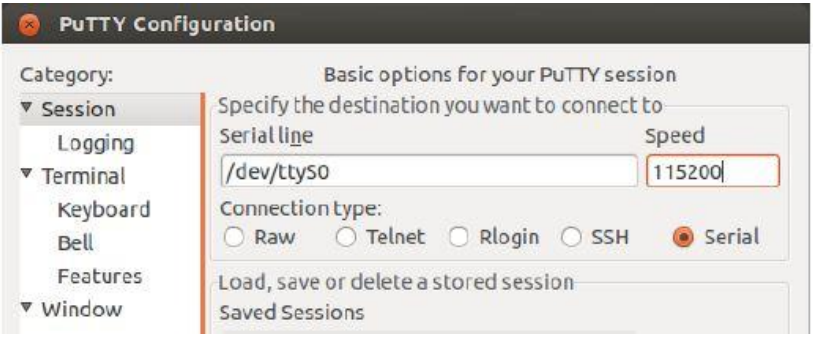
b. Data Bits = 8

c. Parity = None

d. Stop Bits = 1

e. Flow Control = None

f. Preferred emulation mode is ANSI



Basic options for your PuTTY session

Specify the destination you want to connect to

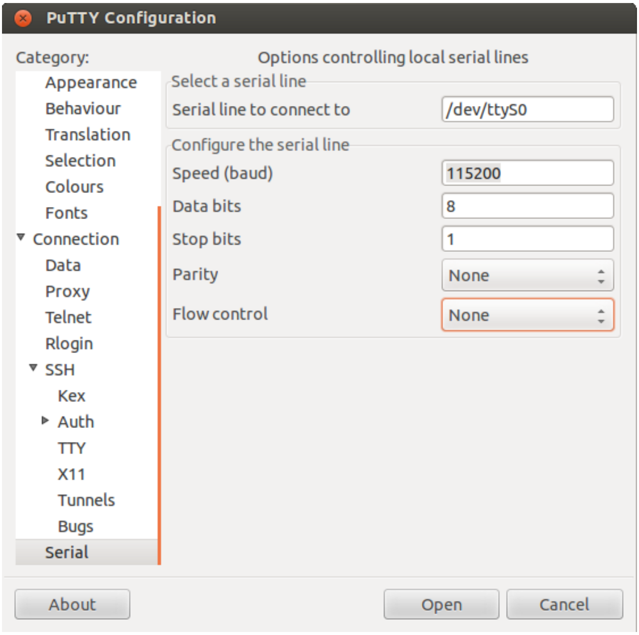
Serial line: /dev/ttyS0

Speed: 115200

Connection type: ○ Raw ○ Telnet ○ Rlogin ○ SSH ● Serial

Load, save or delete a stored session

Saved Sessions



Options controlling local serial lines

Select a serial line

Serial line to connect to: /dev/ttyS0

Configure the serial line

Speed (baud): 115200

Data bits: 8

Stop bits: 1

Parity: None

Flow control: None

4. Power on the target (board).
5. Plug the 2.1mm circular connector on the power supply into the platform 5V DC input. On each of the LAN ports, one LED will be lit.
6. The target (board) will start the boot process. Progress can be observed on the host computer terminal emulator.

Continue with the procedures in this document to set up the software.

2.3.2 Wired Ethernet Connection

The IDP runtime system implements a gateway function that assumes the Ethernet eth0 interface provides a WAN connection, and will attempt to obtain an IP address from a DHCP server in this interface.

There will be a delay in booting when a DHCP server is not present. This may range from several seconds to several minutes.

If you choose not to provide a DHCP service, then an IP address can be statically assigned after the system has booted.

Note: The onboard wireless LAN is statically defined to use the 192.168.1.0 subnet.

2.3.3 Wireless Ethernet Connection

After the system has booted, the IDP gateway will broadcast a wireless LAN with SSID IDPDK-*xxxx* (where xxxx is the last 4 digits of the MAC address of the wireless network card).

To find the last 4 digits of the MAC address of the wireless network card, issue the Linux command: ifconfig wlan0 from the Target System command line. The MAC address will be listed in ifconfig wlan0 output as HWaddr. For example: HWaddr 00:0F:20:CF:8B:42

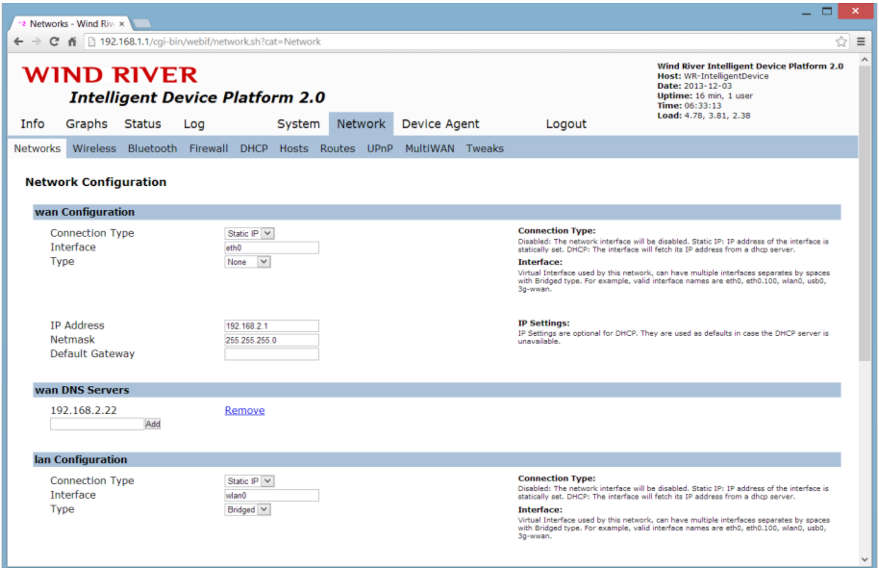
In this case, the last four digits of the MAC address are: 8B42, and the Target System would broadcast a SSID of: IDPDK-8B42.

You may connect to this local network using the password: windriveridp

Once connected, you can access https://192.168.1.1 for configurations.
Login in (user: **admin**, password: **admin)** and go to the **Configuration** tab to configure your system.

To configure a static IP address for the WAN interface, go to the **Network** tab.

Note: If you choose to create a static IP configuration, you must also statically define the WAN DNS server. Refer to the following screenshot.



Wind River Intelligent Device Platform 2.0
Host: WR-IntelligentDevice
Date: 2013-12-03
Uptime: 16 min, 1 user
Time: 06:33:13
Load: 4.78, 3.81, 2.38

Network Configuration

wan Configuration

Connection Type: Static IP
Interface: eth0
Type: None

IP Address: 192.168.2.1
Netmask: 255.255.255.0
Default Gateway:

Connection Type:
Disabled: The network interface will be disabled. Static IP: IP address of the interface is statically set. DHCP: The interface will fetch its IP address from a dhcp server.

Interface:
Virtual Interface used by this network, can have multiple interfaces separates by spaces with Bridged type. For example, valid interface names are eth0, eth0.100, wlan0, usb0, 3g-wwan.

IP Settings:
IP Settings are optional for DHCP. They are used as defaults in case the DHCP server is unavailable.

wan DNS Servers

192.168.2.22

lan Configuration

Connection Type: Static IP
Interface: wlan0
Type: Bridged

Connection Type:
Disabled: The network interface will be disabled. Static IP: IP address of the interface is statically set. DHCP: The interface will fetch its IP address from a dhcp server.

Interface:
Virtual Interface used by this network, can have multiple interfaces separates by spaces with Bridged type. For example, valid interface names are eth0, eth0.100, wlan0, usb0, 3g-wwan.

Chapter 3

Software Features

3.1 Secure Package Management

This section describes features that are included with the AAEON Quark™ SoC X1000 Software package to enable board-specific functions.

The Secure Package Management feature adds secure package management to your target (board). It uses IMA Appraisal to prevent loading applications and libraries without authorized signatures.

A key with authorized signatures is needed to run the application.

ex.

```
evmctl ima_sign ~/Application vendor-private.pem
```

3.2 McAfee Application Control

McAfee Embedded Control: Uses dynamic whitelisting to ensure only trusted applications are allowed on servers and clients.

Refer to the McAfee Product Guide and Release Notes for customization details.

3.2.1 Layers structure

IDP provides a McAfee layer that lets you configure McAfee embedded products for the Wind River Linux target platform. McAfee embedded control (MEC) provides the following capabilities in Wind River Linux target platforms:

- Code and Application Protection: Lets only whitelisted programs (binary, executables, scripts) run. This stops malicious programs from installing and functioning on the system.
- Tamper Proofing for whitelisted programs: Files cannot be modified on the disk. Write and read protection applicable to all types of files, including data files, configuration files, directories, or volumes
- Dynamic Whitelisting: Eliminates the need to manually maintain your list of authorized applications. This feature lets you manage and update whitelisted files.

In this lab you can perform the following tasks:

- Integrate MEC into your Wind River Linux environment
- Explore how MEC manages the inventory of executables, configurations, operation modes, and logging
- Enable McAfee embedded control
- Observe how the MEC code and application protection feature works
- Use the MEC updater component
- Verify the MEC write/read protection feature
- Use MEC update mode

3.2.2 Exploring McAfee Embedded Control

In this section you will explore how McAfee embedded control (MEC) integrates into Wind River Linux and how MEC manages your system.

1. On the target (board) console, as the MEC administrator, execute the following command to confirm that the MEC RPM is in the image running on the target (board).

```
# rpm -qa | grep solidcore
solidcores3-6.1.0_40028-r0.intel_quark
```

2. Execute the following command to confirm MEC application control service (scsrvc) is running.

```
# ps -aef | grep scsrvc
root     4140        1        0   14:59   ?        00:00:00
/usr/local/mcafee/solidcore/bin/scsrvc
root     4143     4140        1   14:59   ?        00:02:10
/usr/local/mcafee/solidcore/bin/scsrvc
root     31693    5281        0   17:44   ttyS1   00:00:00    grep scsrvc
```

3. Execute the following command to display the help menu.

```
# sadmin help
Copyright 2008-2014 McAfee, Inc. All Rights Reserved.
Usage: sadmin <COMMAND> [options] [arguments]

Sadmin is the command line interface to administer McAfee Solidifier.
```

4. Execute the following command to review the list of all application control

features and their status (enabled or disabled).

```
# sadmin features -d
```

Note the following aspects of the MEC features:

- The feature deny-exec prevents unauthorized or unknown binaries from executing. It is based on whitelisting technology, which only allows binaries on the whitelist to execute.
- The feature script-auth is like deny-exec, but for scripts — only whitelisted script files can execute.
- The feature deny-write provides tamper-proofing to protect data files (for example, configuration files). Unlike the deny-exec and script-auth features (which rely on a whitelist), the **deny-write** feature is rules-based. The MEC configuration file (**solidcore.conf**) records the rules.
- The feature **deny-read** provides tamper-proofing to prevent reading of critical files.
- The feature deny-read is also rule based (like **deny-write**) — the MEC configuration file (solidcore.conf) records the rules. This feature is disabled by default.
- The feature integrity protects MEC data and files from modification, renaming, or deletion.

5. As the MEC administrator, execute the following command to check the status of McAfee embedded control on your target (board).

```
# sadmin status
```

Observe that the status is Unsolidified.

The following table describes the fields and their meaning.

Field	Description
McAfee Solidifier	Specifies the operational mode of

	application control
McAfee Solidifier on reboot	Specifies the operational mode of application control after a system restart
ePO Managed	Displays the connectivity status of application control with McAfee ePO. In a standalone configuration, this status is No.
Local CLI access	Displays the status (lockdown or recovered) of the local CLI. In standalone configuration, this status is Recovered.
[fstype]	Displays the supported file systems for a volume
[status]	Displays the current whitelist status for all the supported volumes on a system. If a volume name is specified, only the whitelist status for that volume Displays.
[driver status]	Displays whether the application control driver is loaded on a volume. If the driver is loaded, the status is attached; otherwise the status is unattached.
[volume]	Displays the volume names

6. Execute the following command to display the log file /usr/local/mcafee/solidcore/log/solidcore.log.

```
# cat /usr/local/mcafee/solidcore/log/solidcore.log
```

7. Execute the following command to display the product configuration file /etc/mcafee/solidcore/solidcore.conf.

```
# cat /etc/mcafee/solidcore/solidcore.conf | more
```

Note that the file includes following rules and configurations:

- The run-time mode
- The run-time mode on next reboot
- The license
- The features installed
- The features enabled
- write protect, read protect, and monitoring rules
- The installation directory
- The log file directory

8. On your host computer, open a new terminal window and start an SSH session to your target (board) as the user wruser. When prompted, enter the password wruser.

```
# ssh wruser@$TARGET_IP
```

NOTE: You will use this new terminal window (where you logged in as the user wruser) as the user terminal to perform general user tasks (like running scripts). In this lab, if an instruction says “as the user”, execute the commands on this console.

9. As the user, execute the following command.

```
$ /usr/sbin/sadmin status
Failed to connect to the McAfee Solidifier Service: Insufficient privileges.
```

On MEC, only the administrator (the user root) can execute McAfee application control commands.

10. As the MEC administrator, execute the following command and set the password to **admin**.

```
# sadmin passwd
```

```
New Password:
Retype Password:
Password changed.
```

The administrator (the user **root**) can enable password protection to restrict execution of critical **sadmin** commands. When password protection is enabled, application control lets critical **sadmin** commands run only when the user enters in the correct password.

11. As the MEC administrator, execute the following command and enter a wrong password twice, then enter the correct password (**admin**).

```
# sadmin features list
```

Application control only executes the command when you entered the correct password.

12. In the rest of this lab you will not use password protection. As the MEC administrator, execute the following command to remove the password protection.

```
# sadmin passwd -d
```

3.3 Exploring Webif

Wind River provides a web-based interface called Webif for managing Wi-Fi connections with Intelligent Device Platform target systems.

3.3.1 Objectives

In this lab you will use Webif (a web browser interface for managing targets) to review and alter the operation of you target (board). During this lab, you will perform the following tasks:

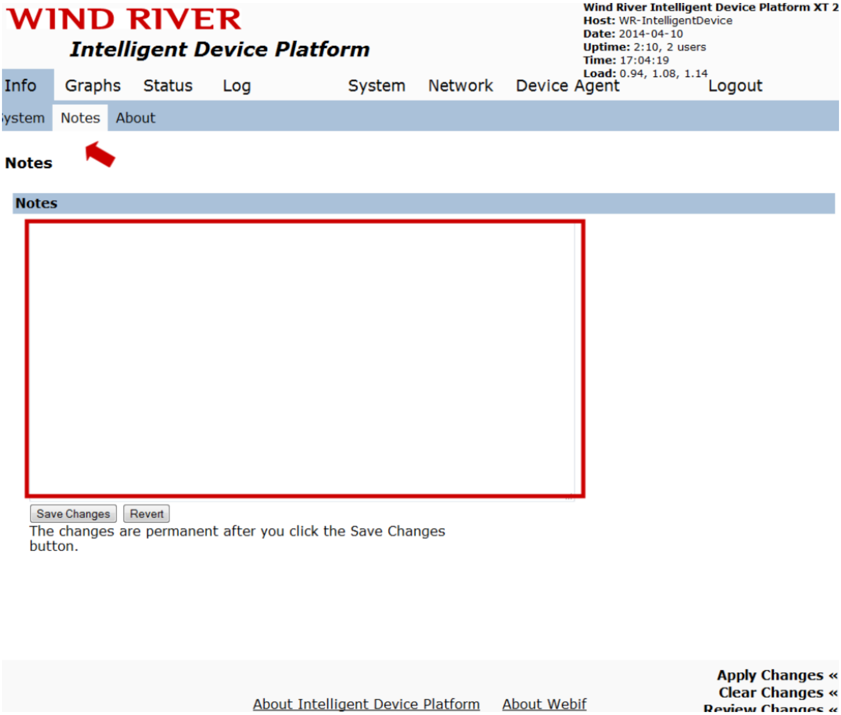
- Connect to the target (board) using Webif
- View the CPU utilization of the target (board)
- Review syslog events
- Add Webif users and give them different views into the target (board)
- Use the **ping** command to verify that the target (board) can connect to various systems

Alter the boot operation of the target system (board)

3.3.2 Working with the Info Page

The Info page is the default landing page for Webif. There are three tabs here, System(the default), Notes, and About. The Notes page lets you store notes about this particular system (you can write anything you want). These notes remain available each time you log in.

1. Click the Notes tab, then enter a note about this system.



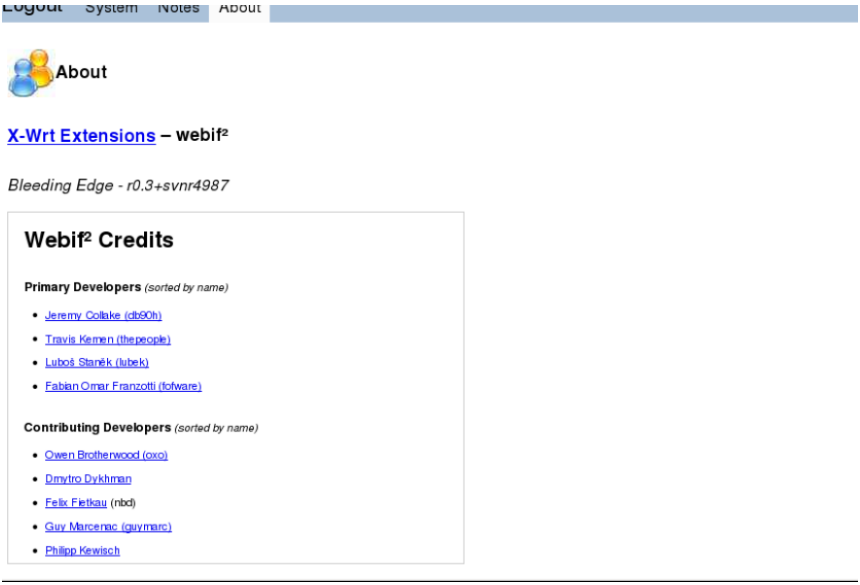
Wind River Intelligent Device Platform XT 2
Host: WR-IntelligentDevice
Date: 2014-04-10
Uptime: 2:10, 2 users
Time: 17:04:19
Load: 0.94, 1.08, 1.14

The changes are permanent after you click the Save Changes button.

2. Click Save Changes to save changes to your notes. Click Revert to remove any changes you have made but have not yet saved.

NOTE: You must click Save Changes to save changes to this page. Webif does not save changes to this page when you click Apply Changes, Clear Changes, or Review Changes.

3. Click the **About** tab. The Webif2 credits scroll automatically after a few seconds.



About

X-Wrt Extensions – webif²

Bleeding Edge - r0.3+svnr4987

Webif² Credits

Primary Developers *(sorted by name)*

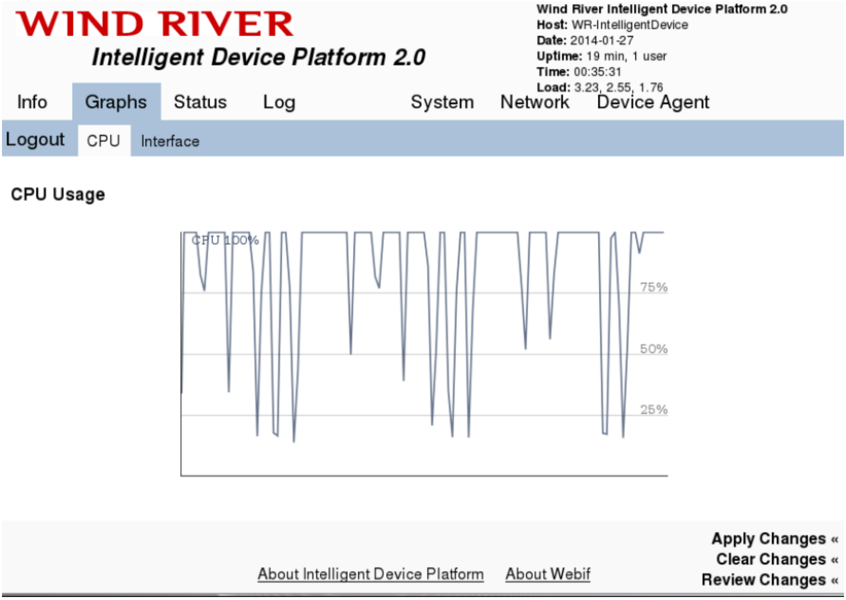
- Jeremy Collake (db90h)
- Travis Kemen (thepeople)
- Luboš Staněk (lubek)
- Fabian Omar Franzotti (fofware)

Contributing Developers *(sorted by name)*

- Owen Brotherwood (oxo)
- Dmytro Dykhman
- Felix Fietkau (nbd)
- Guy Marcenac (guymarc)
- Philipp Kewisch

3.3.3 Working with the Graphs Page

1. Click the **Graphs** tab. The page has two sub-tabs, **CPU** (default) and **Interfaces**. It takes a few seconds before the page displays data. The CPU usage varies depending on the processes and tasks running on your target (board). If you navigate away from this page then return, the graph displays new data beginning from the left margin.



Wind River Intelligent Device Platform 2.0
Host: WR-IntelligentDevice
Date: 2014-01-27
Uptime: 19 min, 1 user
Time: 00:35:31
Load: 3.23, 2.55, 1.76

CPU Usage

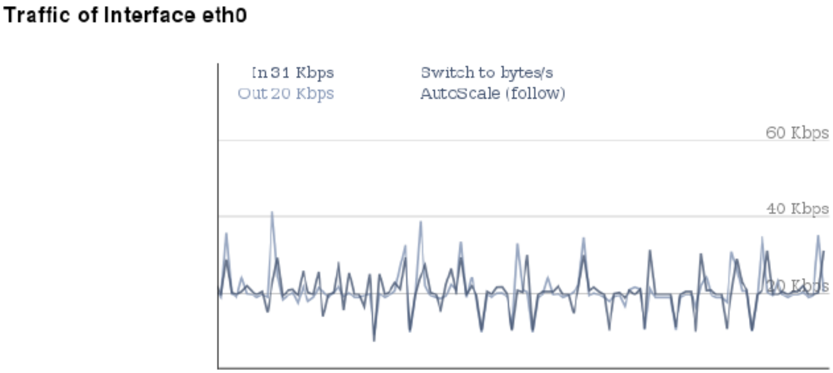
2. Click the **Interfaces** tab. This is a tall page that displays a graph for each network interface. Scroll to see the other interfaces.

3. On the target (board) console, execute the following command to generate some network traffic.

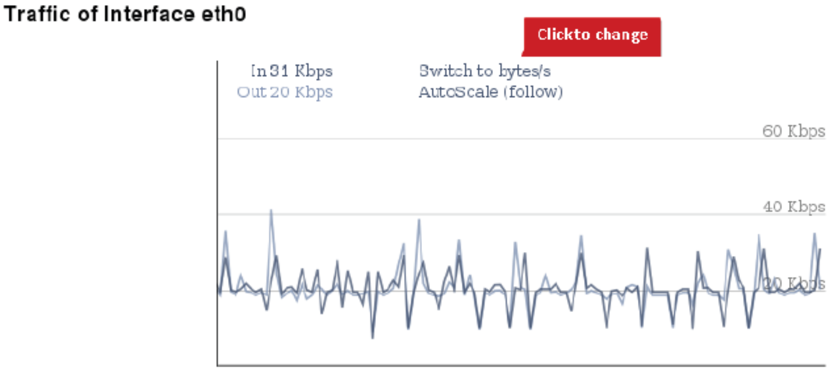
```
# ping -c 5 $HOST_IP -s 64000
```

4. On the Webif page on your host computer, on the **Graphs** > **Interface** tab,

watch the Traffic of Interface eth0 graph change.


Traffic of Interface eth0

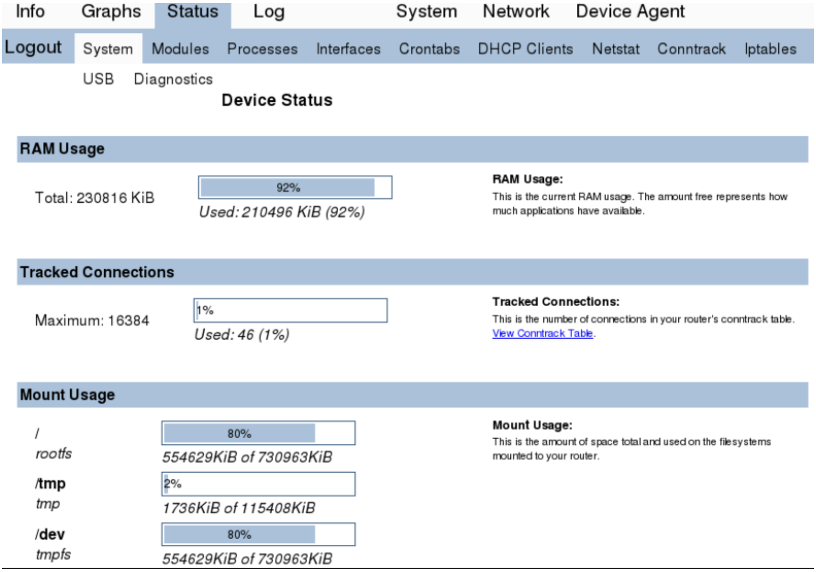
5. You can change the scale of each graph on the **Graphs** > **Interface** tab. Click **Switch to bytes/s** to change the scale from **Kbps** (kilobits per second) to **KB/s** (kilobytes per second). You can switch back and forth as you like.


Traffic of Interface eth0

Clickto change

3.3.4 Working with the Status Page

1. Click the **Status** tab. The **System** sub-tab displays the total space and available space on each mount point, as well as the memory usage and tracked connections. Under the Tracked Connections section, click **View Conntrack Table** to display additional information about your tracked connections (on the **Status** > **Conntrack** tab).



Device Status

RAM Usage

Total: 230816 KiB

92%

Used: 210496 KiB (92%)

RAM Usage:
This is the current RAM usage. The amount free represents how much applications have available.

Tracked Connections

Maximum: 16384

1%

Used: 46 (1%)

Tracked Connections:
This is the number of connections in your router's conntrack table. View Conntrack Table.

Mount Usage

/
rootfs

80%

554629KiB of 730963KiB

/tmp
tmp

2%

1736KiB of 115408KiB

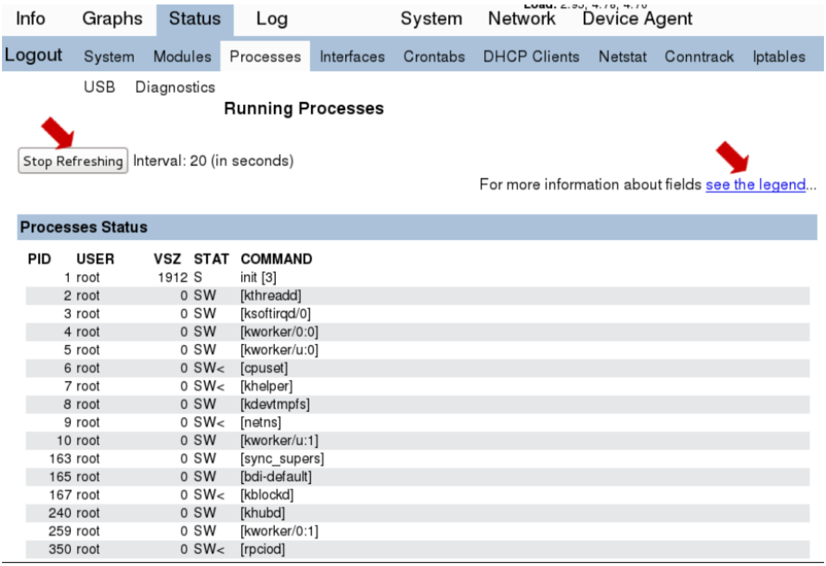
/dev
tmpfs

80%

554629KiB of 730963KiB

Mount Usage:
This is the amount of space total and used on the filesystems mounted to your router.

2. Click the **Processes** tab to display a current list of processes running on the target (board). The page refreshes every 20 seconds unless you click **Stop Refreshing**. Click to **see the legend** to display a legend that describes processes states.



Running Processes

Interval: 20 (in seconds)

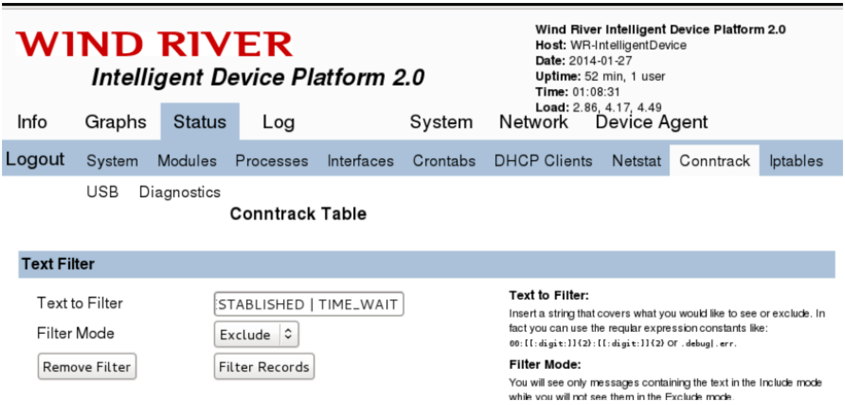
For more information about fields see the legend...

Processes Status

PID	USER	VSZ	STAT	COMMAND
1	root	1912	S	init [3]
2	root	0	SW	[kthreadd]
3	root	0	SW	[ksoftirqd/0]
4	root	0	SW	[kworker/0:0]
5	root	0	SW	[kworker/u:0]
6	root	0	SW<	[cpuset]
7	root	0	SW<	[khelper]
8	root	0	SW	[kdevtmpfs]
9	root	0	SW<	[netns]
10	root	0	SW	[kworker/u:1]
163	root	0	SW	[sync_supers]
165	root	0	SW	[bdi-default]
167	root	0	SW<	[kblockd]
240	root	0	SW	[khubd]
259	root	0	SW	[kworker/0:1]
350	root	0	SW<	[rpciod]

3. Click the **Conntrack** sub-tab to display the currently tracked connections. You can filter out data to focus on the issue you want to resolve.

   In the **Text to Filte**r field, enter **ESTABLISHED | TIME_WAIT** and in the **Filter Mode** field select **Exclude**, then click **Filter Records** to filter these connections out of the display. A subset of the records displays. Verify if the pattern match is case-sensitive.



Wind River Intelligent Device Platform 2.0
Host: WR-IntelligentDevice
Date: 2014-01-27
Uptime: 52 min, 1 user
Time: 01:08:31
Load: 2.86, 4.17, 4.49

Conntrack Table

Text Filter

Text to Filter: STABLISHED | TIME_WAIT

Filter Mode: Exclude

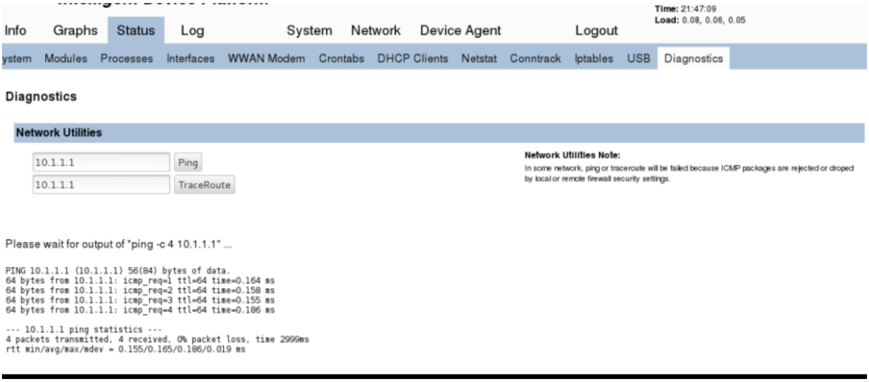
Text to Filter:

Insert a string that covers what you would like to see or exclude. In fact you can use the reqular expression constants like: `00:[[:digit:]]{2}:[[:digit:]]{2}` or `.debug|.err.`

Filter Mode:

You will see only messages containing the text in the Include mode while you will not see them in the Exclude mode.

4. Click the **Diagnostics** sub-tab to run the **ping** and **traceroute** commands for network diagnosis. In the field to the left of the **Ping** or **TraceRoute** button, enter *$HOST_IP* (The IP address of your host computer), then click the button.



Time: 21:47:09
Load: 0.08, 0.06, 0.05

Diagnostics

Network Utilities

10.1.1.1 Ping

10.1.1.1 TraceRoute

Network Utilities Note:

In some network, ping or traceroute will be failed because ICMP packages are rejected or droped by local or remote firewall security settings.

Please wait for output of "ping -c 4 10.1.1.1" ...

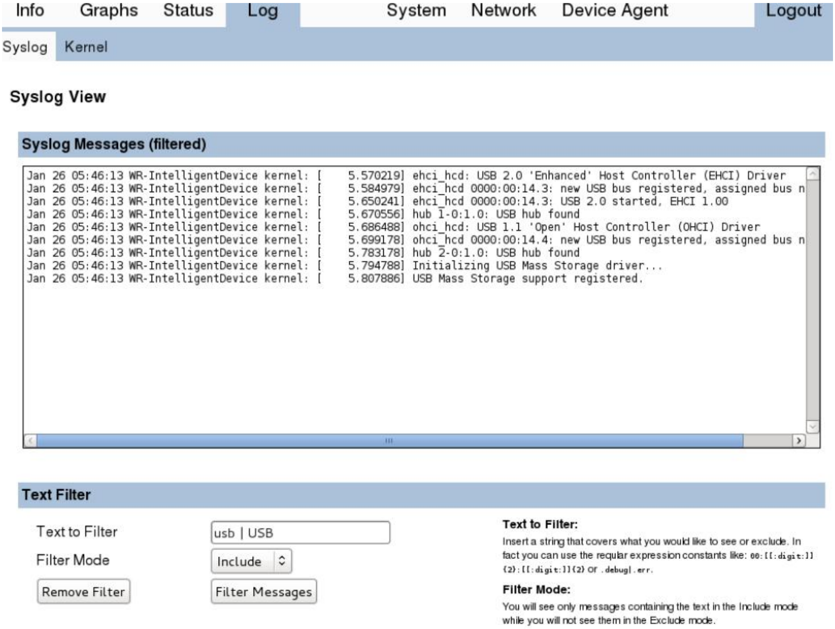
```
PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_req=1 ttl=64 time=0.164 ms
64 bytes from 10.1.1.1: icmp_req=2 ttl=64 time=0.158 ms
64 bytes from 10.1.1.1: icmp_req=3 ttl=64 time=0.155 ms
64 bytes from 10.1.1.1: icmp_req=4 ttl=64 time=0.186 ms

--- 10.1.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.155/0.165/0.186/0.019 ms
```

Note: You can ping and traceroute any domain as long as internet access is available. Internet access will not be available if you are in a Live-Remote class.

3.3.5 Working with the Log Page

1. Click the **Log** tab. The initial view is the **Syslog** sub-tab, which displays the syslog file. You can use the Text Filter section to filter in or out content that you do or do not want to see in the log.

2. In the **Text to Filter** field, enter **usb | USB**, in the **Filter Mode** field select **Include**, then click **Filter Messages** to find all messages in syslog related to USB.



Syslog View

Syslog Messages (filtered)

```
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.570219] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.584979] ehci_hcd 0000:00:14.3: new USB bus registered, assigned bus n
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.650241] ehci_hcd 0000:00:14.3: USB 2.0 started, EHCI 1.00
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.670556] hub 1-0:1.0: USB hub found
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.686488] ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.699178] ohci_hcd 0000:00:14.4: new USB bus registered, assigned bus n
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.783178] hub 2-0:1.0: USB hub found
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.794788] Initializing USB Mass Storage driver...
Jan 26 05:46:13 WR-IntelligentDevice kernel: [    5.807886] USB Mass Storage support registered.
```

Text Filter

Text to Filter: usb | USB

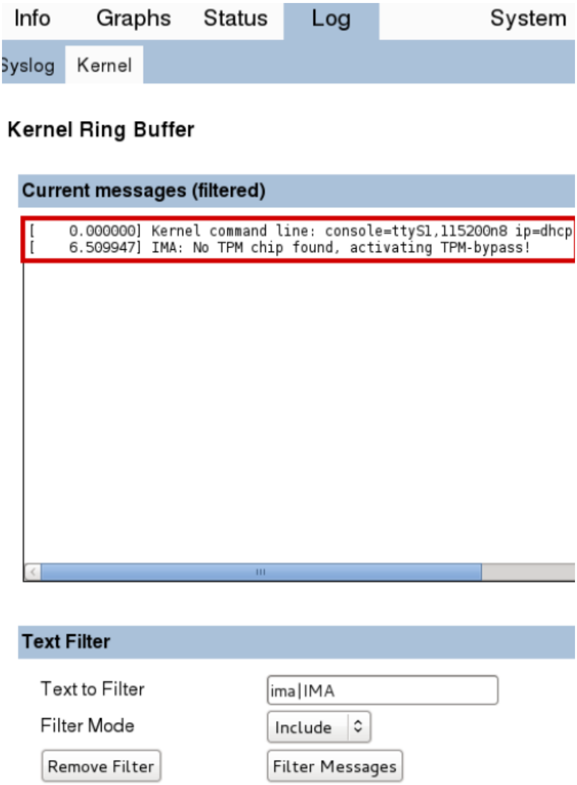
Filter Mode: Include

Remove Filter · Filter Messages

Text to Filter:
Insert a string that covers what you would like to see or exclude. In fact you can use the reqular expression constants like: `00:[[:digit:]]{2}:[[:digit:]]{2}` or `.debug|.err`.

Filter Mode:
You will see only messages containing the text in the Include mode while you will not see them in the Exclude mode.

3. Click the **Kernel** sub-tab and notice that the messages are similar to those in the **Syslog** sub-tab, with the same filtering ability. Filter for **IMA | ima** and observe that TPM is not supported.



Kernel Ring Buffer

Current messages (filtered)

```
[    0.000000] Kernel command line: console=ttyS1,115200n8 ip=dhcp
[    6.509947] IMA: No TPM chip found, activating TPM-bypass!
```

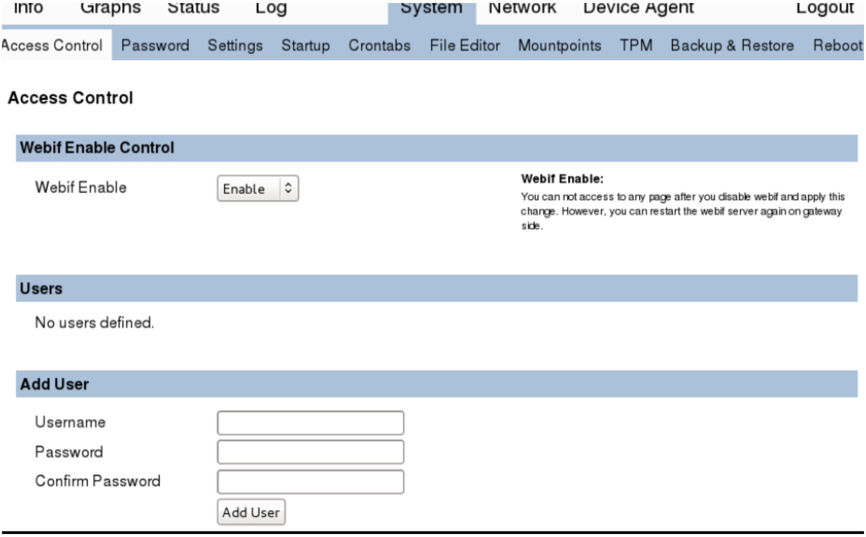
Text Filter

Text to Filter: ima|IMA

Filter Mode: Include

3.3.6 Working with the System Page

1. Click the **System** tab. The default **Access Control** sub-tab lets you add, modify, and remove Webif users to control who can use different pages and tabs within the Webif program. Note that Webif users are not system user log in names.



Access Control

Webif Enable Control

Webif Enable: Enable

Webif Enable:
You can not access to any page after you disable webif and apply this change. However, you can restart the webif server again on gateway side.

Users

No users defined.

Add User

Username

Password

Confirm Password

Add User

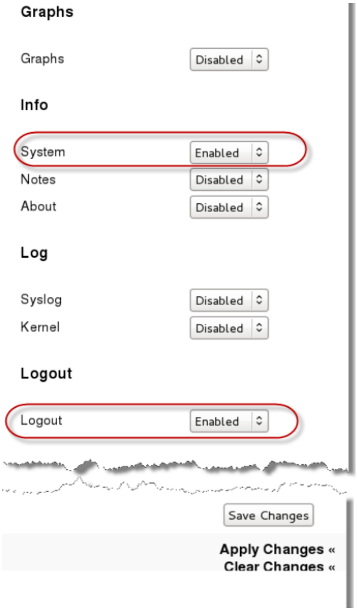
NOTE: Do not change the **Webif Enable** field from **Enable**. If you disable this field, you will lose the Webif connection to the target (board), and you must restart Webif from the target (board).

2. In the **Username** field, enter **Testuser**, in the **Password** field enter **Testpass** and re-enter that password in the **Confirm Password** field, then click **Add User** to add that user to the Webif user database.

3. Give the user **Testuser** access to some of the Webif pages. Scroll down the Access Control sub-tab to configure the following settings, then scroll to the

bottom of the page and click **Save Changes**. After the screen refreshes, scroll to the bottom again and click **Apply Changes**.

- In the Info section, in the **System** field, select **Enabled**.
- In the Logout section, in the **Logout** field, select **Enabled**.



Graphs

Graphs Disabled

Info

System Enabled
Notes Disabled
About Disabled

Log

Syslog Disabled
Kernel Disabled

Logout

Logout Enabled

NOTE: You must click on both **Save Changes** and **Apply Changes** for your changes to take effect.

4. Close the browser.

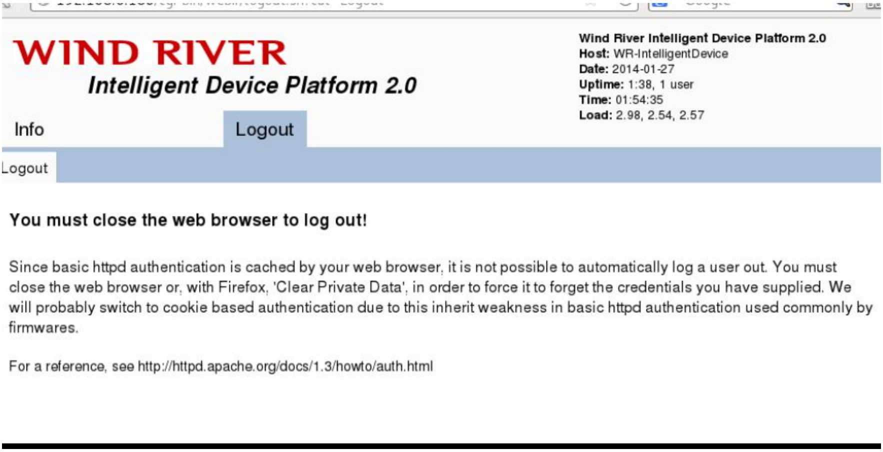
5. Start another browser session then connect to the target (board), but log in as the user **Testuser**. Could you log in? How does the display differ from before?

6. Close the browser

3.3.7 Logout Page

1. Start a browser session and log in as the user **admin**.

2. Click the **Logout** tab, then close the browser. This is the recommended procedure to disconnect from the target system (board).



Wind River Intelligent Device Platform 2.0
Host: WR-IntelligentDevice
Date: 2014-01-27
Uptime: 1:38, 1 user
Time: 01:54:35
Load: 2.98, 2.54, 2.57

You must close the web browser to log out!

Since basic httpd authentication is cached by your web browser, it is not possible to automatically log a user out. You must close the web browser or, with Firefox, 'Clear Private Data', in order to force it to forget the credentials you have supplied. We will probably switch to cookie based authentication due to this inherit weakness in basic httpd authentication used commonly by firmwares.

For a reference, see http://httpd.apache.org/docs/1.3/howto/auth.html

Chapter 4

Quark™ SoC X1000 Drivers

4.1 Overview

System on a Chip in the context of AAEON Quark™ SoC X1000 refers to peripheral hardware south of the host bridge interface. SoC software drivers bind the hardware interfaces into standard Linux* sub-systems. Linux* kernel baseline of 3.8.7 (or higher) is required to ensure proper integration and compatibility of upstream reused kernel drivers.

4.2 Hardware Interface and Drivers

The table below lists the hardware interface implemented on AAEON Quark™ SoC X1000 and identifies whether the associated driver is one of the following:

- Standard (unmodified), off-the-shelf driver
- Modified version of off-the-shelf driver, enhanced to enable AAEON Quark™ SoC X1000 specific features

Note: Refer to the software sources to determine the complete list of modified or added files as compared to the Linux* kernel baseline 3.8.7.

- Created to be AAEON Quark™ SoC X1000 specific

AAEON Quark™ SoC X1000 Hardware Interfaces and Drivers

Hardware Interface	Standard Linux* Driver	Modified Linux* Driver	AAEON Quark™ SoC X1000 Specific Driver
USB OHCI Controller Interface	X		
USB 2.0 EHCI Controller Interface	X		
USB Device Interface		X†	
SD/MMC Controller Interface	X		
UART + DMA Interface		X†	

SPI Master Interface		X	
I²C Master Interface	X		
I²C/GPIO Interface			X
Ethernet Interface		X	

† PCI vendor/device identifiers added for AAEON Quark™ SoC X1000.

NOTE: Refer to the X1000 Drivers section of the Software Developer's Manual for Linux guide for details.

4.3 Expansion Drivers

This section describes drivers that are included with the Intel® Quark™ SoC X1000 Software package to enable board-specific functionality.

- AD7298 Driver
- Bluetooth* Driver (requires mini-PCIe card)
- Wi-Fi* Driver (requires mini-PCIe card)
- 3G Modem Driver (requires mini-PCIe card)

4.3.1 AD7298 Driver

The Analog Devices* AD7298 is a 12-bit, low power, 8-channel, successive approximation ADC with an internal temperature sensor. The LS-ADC does not provide a user-space interface directly, it is provided by the IIO subsystem in the Linux* kernel.

The ADC registers with the IIO subsystem as an IIO ADC device driver. As such, it makes calls to functions on the IIO kernel API and provides callbacks which can be used by the IIO subsystem to invoke driver operations.

To load the drivers for the AD7298, perform the following sequence:

- Enable GPIO driver:
  ```
  modprobe intel_qrk_gip
  modprobe gpio_sch
  ```
- Enable IIO support:
  ```
  modprobe industrialio
  ```
- Enable SPI driver:
  ```
  modprobe spi-pxa2xx
  ```
- Enable AD7298 driver:
  ```
  modprobe ad7298
  ```

After the driver loading sequence is complete, the AD7298 driver enables the following data points via the Industrial I/O (IIO) kernel API directly read from the ADC chip.

Refer to the AD7298 Driver section of the Software Developer's Manual for Linux guide for details.

4.3.2 Bluetooth* Driver

Bluetooth functionality is provided by a mini-PCIe card connected to the mini-PCIe slot on the platform. The following cards have been validated with the AAEON Quark™ SoC X1000 Software:

- Intel® Centrino® Wireless-N 135 card
- Intel® Centrino® Advanced-N 6205 Wi-Fi Radio Module (Dual Band Wi-Fi, 2.4 and 5 GHz)

The following drivers must be loaded to enable USB-bluetooth components:

```
modprobe ehci-hcd
modprobe ohci-hcd
modprobe ehci-pci
modprobe btusbl
```

Once loaded, the sysfs entry below should appear:

```
/sys/module/Bluetooth
```

The following user-space components are required:

bluetoothd
hciconfig
hcitool

Refer to the **Bluetooth Driver** section of the Software Developer's Manual for Linux guide for details.

4.3.3 Wi-Fi* Driver

Wi-Fi functionality is provided by a mini-PCIe card connected to the mini-PCIe slot. The Intel® Centrino® Advanced-N 6205 Wi-Fi Radio Module (Dual Band Wi-Fi, 2.4 and 5 GHz) has been validated with the AAEON Quark™ SoC X1000 Software.

To load a driver for the Intel® Centrino® Advanced-N 6205 Wi-Fi Radio Module, type the following command:

```
modprobe iwlwifi
```

After a successful load of this driver, the following sysfs path is available:

```
/sys/class/net/wlan0
```

Refer to the Wi-Fi* Driver section of the Software Developer's Manual for Linux guide for details.

4.3.4 3G Modem Driver

GSM/3G communications functionality can be provided by a mini-PCIe card connected to the mini-PCIe slot. The Telit* HE910 mini-PCIe module (specifically, the functionality for GSM Voice and SMS communications, and HSPA+ data communications) has been validated with the Intel® Quark™ SoC X1000 Software.

Driver Requirements:

- Telit* HE910 requires USB2.0 support in kernel
- Telit* HE910 requires PPP (point-to-point protocol) support in kernel
- Use of active GPS antenna needs external circuit for powering antenna's amplifier

Software tool requirements:

- minicom - for running scripts

  Can be compiled as ipk package
- microcom - handy for executing simple AT commands

  Microcom is a part of busybox package.

  If it is not installed, it can be enabled in yocto using the command:

  bitbake busybox -c menuconfig

  then re-installed as ipk package.
- pppd - Point-to-point protocol

  ppp is used for data packet connection. It can be enabled in yocto as an image feature "ppp"

To load the drivers, perform the following sequence:

- Enable USB controllers:

  modprobe ehci-hcd

  modprobe ohci-hcd

  modprobe ehci-pci

- Enable Communication Device Class Abstract Control Model interface:

  modprobe cdc-acm

Refer to the 3G Modem Driver section of the Software Developer’s Manual for Linux guide for details.