Aculab Prosody X card

Card network integration and administration guide

Revision 6.7.0

PROPRIETARY INFORMATION

Document Revision

Rev	Date	By	Detail
6.4	17-01-07	djl	Initial release for V6.4 telephony software support
6.5	14-01-13	mb	Amendment for V6.4 telephony software support
6.5.1	22-01-13	ebj	Reformatted and updated to corporate fonts.
6.6.0	27-04-15	pgd	Align with 6.6 distribution
6.7.0	11-04-17	pgd	Add DHCP advice

CONTENTS

1 Card boot up and discovery 5
1.1 Card boot 5
1.2 Card discovery 6
1.3 Card lists 6
1.3.1 Prosody X Settings Card List 6
1.3.2 Main card list 7
1.4 Troubleshooting 8

2 Network setup considerations 9
2.1 Frequently asked questions 9
2.1.1 How do I connect up my network? 9
2.1.2 Have I got enough Ethernet bandwidth? 9
2.1.3 Can packet fragmentation be a problem for Prosody X? 9
2.1.4 How do I choose IP addresses for Prosody X? 9
2.2 Network connectivity 10
2.2.1 Background 10
2.2.2 Ethernet hubs 10
2.2.3 Ethernet switches 10
2.2.4 Routers 10
2.3 Network bandwidth 10
2.3.1 Available Ethernet bandwidth 11
2.3.2 Symptoms of reaching bandwidth limits 11
2.4 IP address choice 12
2.4.1 What IP addresses do I need to set up and their interrelation? 12
2.4.2 Can I use DHCP to provide the IPv4 addresses? 12
2.4.3 IPv4 address and subnet mask 12
2.4.4 What IPv4 addresses should I choose? 13

3 Security considerations 14
3.1 Configuring firewalls 14
3.1.1 Ports to block 14
3.1.2 Ports to leave open 15
3.2 Card security usability 15
3.2.1 Choosing the key 15
3.2.2 Entering the key 15
3.2.2.1 V6 framework 15
3.2.2.2 Low level functions and applications 15
3.3 Card security low level procedures 16
3.3.1 Card bootstrap 16
3.3.2 Setting the key 16
3.3.3 Using the key 16
3.3.4 Security assumptions 17
3.3.4.1 Security of local switched network 17
3.3.4.2 Handling of denial of service attacks 17
3.3.4.3 No physical access to hardware 17
3.3.4.4 Access to run time security key 17
3.3.4.5 Access to security mechanism 18
3.3.5 Summary 18
3.4 Vulnerabilities 18
3.4.1 Blind attackers 18
3.4.2 Accidental or deliberate use of Aculab software by a third party 18
3.4.2.1 Malicious host on local switched network 18
3.4.2.2 Second host on different network 18

3.4.3 Sighted attackers 19
3.4.3.1 Attacker on same switched network 19
3.4.3.2 Attacker on same or different switched network 19
3.4.4 Denial Of Service attack 20

4 Security Features 21
4.1 Secure host mode 21
4.1.1 Enabling secure host mode 21
4.1.2 Disabling secure host mode 21
4.1.3 Network configuration considerations 22

5 Flash upgrades 23
5.1 Introduction to flash upgrade 23
5.1.1 What is flash upgrade? 23
5.1.2 Why is it needed? 23
5.1.3 When should it be carried out? 23
5.2 Flashing Prosody X cards 23
5.2.1 How to discover the version of flash image on your card 23
5.2.2 Flashing the card using prosody_ip_card_flash 24
5.2.3 Diagnosing problems following a flash upgrade 25

1 Card boot up and discovery

Prosody X system elements make take the form of Prosody X PCIe cards or Prosody X media processing chassis. A Prosody X PCIe card may be viewed as a networked computer, a Network Interface Controller (NIC) and an external Ethernet socket all integrated on a card and interconnected via an on-card Ethernet switch. A Prosody X media processing chassis is similar but omits the NIC element. This document explains how such cards and chassis may be set up and integrated into a IP network environment. It also explains how cards are booted and discovered, along with how they appear in the lists of cards as seen in Aculab Configuration Tool (ACT) or more generally as visible through the Aculab resource management API. It covers the problems that may be encountered and where to look if things do not 'just work'. The same mechanisms are used for both Prosody X PCIe cards and media processing chassis (the generic term Prosody X card is used in the rest of this document to refer to both).

Prosody X PCIe cards and the High availability media processing chassis are IPv6 and IPv4 capable, whilst the 1U enterprise chassis is restricted to IPv4. In this document, where reference is made to IPv6, this is only applicable to IPv6 capable products.

Instructions on how to install a Prosody PCIe X card are contained in the ***Aculab Prosody X PCIe R3 card installation guide***.

This document is aligned with the 6.7 distribution of the V6 Telephony software and the products that are supported by this distribution.

1.1 Card boot

The Power PC on the Prosody X card boots from flash memory into a basic state in which it is capable of Ethernet communication. To be useful, the card now needs to be booted fully from a host. In order to begin this process, the card broadcasts low-level protocol 'boot me' packets which request an Aculab host to boot it fully. An Aculab host will reply to such a request if the card is in its Prosody X card list (see below) and specified as one it can boot. A 'boot me' packet contains the serial number of the Prosody X card, card MAC address information and various sub-fit details such as the numbers of telephony ports and media DSPs.

When a host replies to a 'boot me' packet from a card, it sends a low-level protocol packet containing sufficient information for the card to obtain an IP set-up. If more than one host replies, the one whose reply is received first will boot the card. Once the card has obtained an IP address, it sends this to the host using the low level protocol and the remainder of the boot sequence is achieved over TCP. This consists of downloading various files from V6 telephony software installation on the host to the card and starting various processes on it. Once the boot sequence has completed the card is usable and will appear in the host's main card list and will be shown as having the state "In Service" in the list displayed on the ACT "Prosody X settings" page.

The boot sequence has to be initiated using a low level protocol because, at its start, the card has no IP address. One result of this is that only hosts on the same switched network as the card can boot it - the low level protocol cannot pass through routers.

1.2 Card discovery

In the case where a host wants to use a card that it has not booted, it is necessary for the host to discover the card.

The way in which this discovery is initiated depends on how the card appears in the host's Prosody X card list. If the card is specified with a static IP address, the host initiates contact with it using TCP. Otherwise, the host does not know the address of the card and must therefore initiate contact with it using a low level protocol. Note that a consequence of this is that, in the latter case, card discovery will work only when the card is on the same switched network as the host.

Initially a Prosody X card will just be shown on the on the ACT "Prosody X settings" page and the various stages occurring during its boot process will be displayed in the Diagnostics/Status columns. Note that it will not appear in the ACT "Card List" page until the boot sequence has completed and the card has been configured and reached the "In Service" state. Note also that, if the ACT has been used to set up configuration of signalling and TiNG firmware downloads, these will occur every time a host has brought a card into the "In Service" state such that it appears in the ACT "Card List". If more than one host has the card specified in its Prosody X card list, then you need to be aware of this behaviour. It is, in general, best to configure the ACT for just one host to download firmware to a card.

1.3 Card lists

1.3.1 Prosody X Settings Card List

This list contains all Prosody X cards that the host would like to appear as "In Service" entries in its main card list, along with information to allow the host to boot or discover each one.

Local cards (those physically installed in the host) are added automatically to the Prosody X card list. The host software makes the addition when it receives a low level protocol 'boot me' packet from a card who's NIC it recognises as one installed in the host. Local cards may be removed from the list but will appear again if the host is rebooted - the assumption is made that, if the card is installed in a particular host, that host will want it to appear in its Prosody X Settings card list.

Remote cards (those contained in other hosts or in media processing chassis) must be added to the Prosody X card list manually. This may be achieved either by using the ACT or prosody_ip_card_mgr command line tool or the Aculab Resource API `acu_register_prosody_ip_card()` function. Similarly, they may be removed either by using the ACT or prosody_ip_card_mgr or the Aculab Resource API `acu_unregister_prosody_ip_card()` function.

Each entry in the Prosody X Settings list specifies:

- If the host can boot the card and, if so, what IP set-up it should give it;
- If the card watchdog should be enabled and with what timeout.

This information is provided either when the card is added to the list (see above) or by using the ACT or the Aculab Resource API `acu_configure_prosody_ip_card()` function to modify an existing entry.

Specifying that a host can boot a card means that the host will try to boot the card if it receives a low level 'boot me' packet from it. It does not mean that the host will always boot the card – just that it will attempt to do so if the opportunity arises. Specifically, if a card is already booted then the parameters provided are used only to discover the card (see the 'Card discovery' section above) and are not imposed on it.

The Power PC on the Prosody X base card contains a watchdog which may be configured as part of the boot process. If enabled, this will restart the card if any of the following occurs for longer than the specified timeout period:

- Certain card processes failing to 'kick' the watchdog
- No host maintaining a TCP connection with the card (i.e. no host having the card in its main card list)

The watchdog may be useful in cases where the card is physically remote from (and likely not on the same switched network as) the host. However, unless such operational considerations oblige, it is normally better not to enable the watchdog so that, in the event of a card failure of some kind, the card state will not be lost on a watchdog restart, and thus it will be possible to retrieve diagnostic data in collaboration with advice from Aculab support.

1.3.2 Main card list

This list contains all Aculab cards that the host can control and therefore consists of entries for any Prosody S servers and the subset of cards in the Prosody X card list that the host can communicate with and that have reached the "In Service" state.

1.4 Troubleshooting

This section covers a number of problems that may occur when booting and discovering cards, and suggests ways of fixing them.

If the card is local, check that its NIC has been set-up correctly and, in particular, has a valid and current IP set-up. If it should have a static IP address, does it have such an address and an appropriate subnet mask? If it should obtain its address via DHCP then is the card's external Ethernet connector plugged into an Ethernet, is there a DHCP server, and has the NIC managed to obtain an address? For each supported OS the IP address checks are done in the same way as for any other NIC in the system.

Check the host's routing table to confirm that outgoing IP packets sent to the card's IP address (or, if it's not yet booted, the address you intend to boot it with) can actually get to the card. For local cards this will generally be via the card's NIC, whose IP address you will know from the previous check. If there is no route for such packets to reach the card, change the routing table so that there is. If there is more than one route then check that each of them is valid. If all such routes are not valid, then communications with the card may be intermittent or non functional.

Has another host booted the card? If so, and the card has been entered in the Prosody X card list with a static IP address, your host will not be able to successfully discover the card without this address matching the current address of the card. The list entry can be viewed, for example, by using the ACT.

If the card can only be accessed from your host via a router (i.e. is on a different switched network) then your host cannot boot the card and another host must do so. Bearing in mind the way in which Prosody X cards are discovered, the card must be entered in the Prosody X card list with a static IP address - this is the only way the host can discover the card without using the low level protocol (and therefore being restricted to the same switched network). Then, is there a route (via the router) for packets to reach the card and was the card booted with an IP set-up that included a router through which it can send packets to your host? Check this by trying to ping the card from your host. If you can't ping the card, your host won't be able to discover it.

2 Network setup considerations

A Prosody X PCIe card consists of a number of elements integrated on a card interconnected with an Ethernet switch. This Ethernet switch has two Ethernet interfaces to the outside world: the NIC (via the host PCIe bus) and the external Ethernet RJ45 socket. Many tasks may be performed via the PCIe bus without connecting the card to an external Ethernet through its RJ45 connector, but for some this connection is required. These typically include IP Telephony media processing and control of the card from a remote host.

In contrast, a Prosody X media processing chassis evidently does not have a PCIe bus link to the host computer, and thus it and host must always be connected to the same Ethernet LAN.

This section provides information about the Ethernet LAN environment and Ethernet settings appropriate for Prosody X cards.

2.1 Frequently asked questions

2.1.1 How do I connect up my network?

If connected to an external Ethernet, Prosody X must be connected to a switched and/or routed full duplex network. Use of Ethernet hubs in any part of this network, which carries Prosody X media or control traffic must be avoided. If routers are used, they must be able to sustain the bandwidth required. To see why, please read the Network connectivity note in section 2.2.

2.1.2 Have I got enough Ethernet bandwidth?

Prosody X uses Ethernet bandwidth for card control and IP Telephony media streaming. In typical systems, the dominant users are the TiNG API and the IP Telephony media streams. Most systems will not run into bandwidth limitations, review the information in section 2.3 for more details of symptoms that might occur if your system does.

2.1.3 Can packet fragmentation be a problem for Prosody X?

Prosody X must be used on networks where little or no TCP packet fragmentation occurs and no UDP fragmentation occurs in path between application and card. Most installations will never encounter fragmentation issues. Normally fragmentation would only occur on end-to-end paths that include low Maximum Transmission Unit (MTU) networks such as X.25 and PPPoE.

2.1.4 How do I choose IP addresses for Prosody X?

Prosody X PCIe cards may be set up to use IPv4, IPv6 or both. With respect to IPv4 a Prosody X PCIe card will need two IP addresses, one for the collection of processing elements on the card (PowerPC, DSPs) and one assigned to the card's integrated NIC. Since the two devices are on the same switched network, the two addresses must always be on the same subnet. Choosing IPv4 addresses is a general networking issue but some guidance is provided in the IP address choice note in section 2.4. For IPv6, the individual processing elements on the card each need their own IPv6 address together with one for the card's integrated NIC (but these are normally automatically assigned by IPv6 address auto configuration).

2.2 Network connectivity

2.2.1 Background

Machines on a local network are interconnected using Ethernet hubs or switches. Local networks are interconnected using routers. Prosody X is designed to work properly on networks containing only full duplex, low latency interconnects. This specifically rules out use of Ethernet hubs and, if present, problems will be encountered as channel counts are increased. Prosody X will not be supported in such configurations. VPNs also should be looked at closely, particularly in terms of latency.

The paragraphs below explain the reasoning behind these statements.

2.2.2 Ethernet hubs

On receiving a packet on a port, an Ethernet hub will immediately transmit that packet on all other ports, the hub being merely a repeater. This means that bandwidth on each port is wasted by the needless transmission of unicast packets that are in fact destined for machines on other ports. It also means that all ports work half duplex - data may only flow in one direction at a time. Data collisions must therefore be avoided, retransmissions are required and the main implication of all this is that theoretical bandwidth is very rarely obtained. This is the source of the idea that a half duplex Ethernet will become saturated at around a third of its theoretical bandwidth. The figure itself is of limited significance because it is heavily dependent on the characteristics of the data on the network. But, the idea is important.

This unpredictable reduction in bandwidth means that hubs are unsuitable for high performance systems like Prosody X.

2.2.3 Ethernet switches

An Ethernet switch inspects the MAC addresses of each incoming packet, stores it and transmits (forwards) it only on the required ports. For unicast packets destined for hosts whose location has been learned by the switch, transmission occurs on only the required port. For other packets, transmission occurs on all ports. In almost all properly functioning networks, packets meeting the first condition predominate by far. Therefore the vast majority of packets flow point to point. This, coupled with the store and forward architecture, means that all ports work full duplex - data can flow in both directions at the same time. Therefore it is possible to approach the theoretical bandwidth of the ports. This is somewhat simplified but, again, it's the idea that's important.

2.2.4 Routers

A router inspects the destination IP address (and maybe UDP/TCP port) of each incoming packet, stores it and transmits (forwards) it only on the required ports. It is conceptually similar to an Ethernet switch, but operating on each packet at a higher level.

It is quite possible to pass the Prosody X media and control paths through a router provided it is capable of sustaining the bandwidths required. This is mentioned only because it's not unusual to see quite low specification boxes set up as routers between networks which, while adequate for general use, may struggle to pass data at approaching wire rate (e.g. 100Mbps).

2.3 Network bandwidth

CAUTION

Prosody X must be used in networks with sufficient available bandwidth for the tasks

it is performing.

2.3.1 Available Ethernet bandwidth

A Prosody X card contains a number of processing elements interconnected by an Ethernet switch. This ethernet switch has a link to external ethernet connector(s) , and in the case of Prosody X PCIe card, a link to the integrated PCIe NIC. The links to this switch and its bandwidth are noted below for each Prosody X card variant:

- Prosody X PCIe card:
  - Ethernet link bandwidth: 1000Mbs full duplex or 100Mbs full duplex depending on jumper configuration (see installation guide)
  - Links into card: External Ethernet connector, Host Ethernet (via PCIe bus)
- Prosody X 1U enterprise chassis
  - Ethernet link bandwidth: 100Mbps full duplex
  - Links into card: External Ethernet connectors
- Prosody X High availability chassis
  - Ethernet link bandwidth: 1000Mbps full duplex
  - Links into card: External Ethernet connectors

Prosody X uses Ethernet bandwidth for control (e.g. call, switch or TiNG APIs) and RTP (used to carry VoIP media) for IP telephony. Of these, the TiNG API and RTP will generally be the dominant users.

2.3.2 Symptoms of reaching bandwidth limits

As the bandwidth limit of an Ethernet is approached, performance degrades in two main ways:

- The proportion of packets lost increases.
- The delays in packet transmission become less predictable. This variability is termed jitter.

The effects of increases in these figures depend on the high level protocols in use:

- IP Telephony: RTP is used to transmit media, generally audio. The receiver's Jitter Buffer will trade delay with the proportion of packets arriving too late to be inserted into the decoded media stream in an attempt to maximise perceived audio quality in terms of both delay and fidelity. Increases in the delay and jitter lead to a reduction in received quality, dependant on the codec type and performance of receiver's Packet Loss Concealment (PLC) algorithm.
- Call and Switch APIs: TCP is used here and, as it's a reliable protocol, increases in the above figures lead to a higher proportion of retransmissions, resulting in more bandwidth being used and slower response times.
- TiNG API: The protocol used here is also reliable, so increases in the above figures lead to higher proportion of retransmissions, resulting in more bandwidth being used and slower response times which may lead to underruns/overruns for play/record type activities.

NOTE

Because the call, switch and TiNG APIs are interactive (with the caller and the application) delays need to be kept small, emphasising the importance of having a high quality low packet loss network which is not operating at or near its maximum bandwidth.

2.4 IP address choice

NOTE

Choosing IP addresses is a general networking issue so, while this note gives some guidance, it is not a substitute for good networking practice and knowledge of the set up of your local network. Choosing IP addresses for Prosody X is much like choosing them for PCs. If in doubt, please consult your network administrator.

2.4.1 What IP addresses do I need to set up and their interrelation?

A Prosody X PCIe card will need two IPv4 addresses, one assigned to its processing elements (Power PC, DSPs) and one assigned to its integrated Network Interface Controller (NIC). If use of IPv6 is enabled, it will need an IPv6 address for its NIC and individual IPv6 addresses for its various processing elements - these are normally assigned through IPv6 address auto configuration.

2.4.2 Can I use DHCP to provide the IPv4 addresses?

Aculab recommends that live production systems do NOT use DHCP (Dynamic Host Configuration Protocol) for IPv4 address assignment. It may be convenient to use DHCP assigned addresses in a test lab environment where IPv4 DHCP address assignment will work provided that the card is connected to a network via its external Ethernet socket, the network has a DHCP server, and that server is configured such that the assigned IPv4 address lease is reliably maintained whilst Prosody X card is operational. If you are going to use DHCP address assignment the next two subsections may be ignored, otherwise please read on.

2.4.3 IPv4 address and subnet mask

An IPv4 address is a 32 bit number usually written as 4 'dot' separated values between 0 and 255 inclusive - e.g. '192.168.1.36'.

A subnet mask is part of the configuration for a network and is a 32 bit number which almost always consists of one or more '1's followed by one or more '0's - e.g. '255.255.255.0' or /24, which are 2 ways of saying the same thing. The IP address and subnet mask are often written in combination - e.g. '192.168.1.36/24'. The subnet is the bitwise AND of the IP address and the subnet mask - e.g. '192.168.1.0' in the previous example.

When a computer wishes to send an IP packet to a remote machine, it calculates which subnet the remote machine is on by bitwise ANDing its IP address with the subnet mask of the network (see above). If that subnet is the same as its own, it will send the packet directly. If it's different then it will send it via a router.

IP addresses, subnet masks, subnets and the relationships between them are described in many books about IP and in many places on the web.

2.4.4 What IPv4 addresses should I choose?

As noted above, a Prosody X card may be viewed as containing a networked computer and a Network Interface Controller (NIC). Each device requires an IP address and, since the two devices are on the same switched network, they must each have the same subnet mask and the two IP addresses must be on the same subnet.

Given that, the rest depends on how your network is set up. There are many possible set-ups so we cover just the more common ones here.

Prosody X card(s) connected to an external Ethernet

For this case we assume that one or more Prosody X cards is installed in a host and all are connected to the same switched network via their external Ethernet sockets. This is the usual case of having many devices all connected to the same switched network and so the usual rule applies. All the devices must have the same subnet mask and all the IP addresses must be on the same subnet. In other words, if there's an existing addressing scheme in use on the switched network, make sure the one you choose fits in with it. If there's not one in use then please see RFC1918, Address Allocation for Private Internets (http://www.faqs.org/rfcs/rfc1918.html).

The less usual aspect of this case occurs if we have more than one Prosody X card: the local host now has more than one NIC connected to the same switched network. If you want to ensure that packets flowing between the local host and a given card's networked computer pass through that card's NIC then manipulation of the local host's routing table will be required. Manipulation of routing tables is somewhat Operating System specific but generally accomplished through using the "route" command. This is a general networking issue.

Prosody X card(s) not connected to an external Ethernet

On each card, we have two devices (the card's networked computer and NIC) connected to the same-switched network. Both devices must have the same IPv4 subnet mask and their IPv4 addresses must be on the same subnet. This switched IPv4 network contains only two devices and, unless the card is connected to an external Ethernet, it will always do so. Therefore it is sensible to use the tightest subnet possible for a network: one of 30 bits (e.g. 192.168.12.30) the 4 remaining bits allowing room for two devices, the zero address and the broadcast address - there's no point in wasting valuable IPv4 addresses.

If we have more than one Prosody X card, a different subnet must be chosen for each. In just the same way, the subnets chosen must not fall within subnets that the host can already 'see'. Not adhering to either of these rules would introduce an ambiguity with the IP addresses chosen referring to more than one device. This is a general networking issue, equivalent to the local host being connected to multiple networks with no router between them.

Please see RFC1918, Address Allocation for Private Internets (http://www.faqs.org/rfcs/rfc1918.html).

3 Security considerations

Prosody X is effectively an IP networked computer running on a card. When deploying any computer on an IP network it is sensible to consider the security implications. This section gives you the information you may need to achieve this.

The following classes of IP network based attack are considered:

- Blind attackers (port scans etc.)
- Accidental or deliberate use of freely downloadable Aculab software by a third party
- Sighted attackers (e.g. a competing company wanting to take a PX card out of service)
- Knowledgeable attackers (e.g. ex-employees)
- Denial of Service (DoS) attacks.

3.1 Configuring firewalls

Any computer connected to an IP network will be more secure from attack if it is placed behind a suitably configured firewall. This section lists how such a firewall should be configured to improve the security for Prosody X

3.1.1 Ports to block

Here is a list of Prosody X PCI TCP and UDP ports which should be blocked against traffic originating from anywhere other than the controlling host machine. Note that the ports may change with future card revisions.

Protocol	Port
TCP	220 (ppcctld), 222 (hdirelay), 2031 (asspmon), 3013 (TDM switch), 6614 (STUN), 8190 (IPTel), 8191 (TRM)
UDP	2030 (cardinfod)
UDP	16384 — 16386, 24576 — 24578, 32768 — 32770, 40960 — 40962 (DSP 0 — 3 debug, ASSP, ASSPmon)

3.1.2 Ports to leave open

Alternatively, all Prosody X TCP and UDP ports apart from those in the list below should be blocked against traffic originating from anywhere other than the controlling host machine. Again this is for security purposes and the DSP ports may change with future card revisions.

Card	Protocol	Port
PCI	UDP	16387 — 17407, 24579 — 25599, 32771 — 33791, 40963 — 41983 (DSP 0 — 3 RTP, RTCP et al)

3.2 Card security usability

3.2.1 Choosing the key

In systems where only one host bootstraps and controls a given Prosody X card, there is no particular restriction on the choice of key.

However, if more than one host is set to bootstrap and/or control a Prosody X card, all such hosts must use the same key for that card. If this rule is not adhered to, only the host that bootstrapped the card will be able to control it and other hosts will fail to connect to it. This is correct behaviour: the card cannot be controlled by a host that doesn't use the same key with which the card was bootstrapped.

3.2.2 Entering the key

3.2.2.1 V6 framework

The card security procedures rely on a security key being entered on the host. This is entered by the user when a card is added to the Prosody X card list, either via the ACT or directly using the V6 resource management API.

The key is stored on the file system in the V6 tree and is thus known to the V6 framework.

3.2.2.2 Low level functions and applications

Certain low level functions and applications are designed to work standalone so they can be run on hosts, which do not have the V6 framework installed. These include the TiNG low-level functions to open Prosody functionality on a card, and the TiNG test and diagnostic applications. Such functions and applications will require the security key to be passed to them when addressing a Prosody X card.

3.3 Card security low level procedures

Please read this section in conjunction with section 1 “Card boot-up and discovery”, which explains the boot and discovery procedures more fully.

3.3.1 Card bootstrap

When a Prosody X card has booted from flash memory it is unaware of it’s IP address or security key. This is by design and allows a card to be placed into any system and “just work”. It will announce its presence by broadcasting low-level protocol boot-me packets. The card will listen for low-level protocol response packets and will allow itself to be booted by the host which sent the first valid one received. Once the card has obtained an IP address, the remainder of the boot and run procedures are performed unicast using IP packets.

The low-level protocol boot-me packets broadcast from the card should be received by every host on the same switched network as the card. The low level protocol and IP packets unicast subsequently by the host or the card should be received only by the card or the host, respectively, provided all of the following are true:

- the integrity of the switched network is maintained (e.g. no port mirroring);
- all Ethernet switches in the path are functioning correctly
- the machines’ locations remain in the cache of each Ethernet switch in the path;
- no Ethernet hub is in the path.

Low-level protocol packets cannot pass through routers or gateways, provided their integrity is maintained, so a Prosody X card can be booted only from the same switched network. IP packets can pass through routers and gateways, thus a Prosody X card can be controlled from the same or a different switched network.

3.3.2 Setting the key

Early in the bootstrap procedure, the host unicasts to the card a low level protocol packet containing the boot security key. This packet, termed the “boot key packet”, is sent clear channel (unencrypted). The boot key is used to prevent any other host controlling the card during the boot procedure. It is made from a combination of the booting host name, the card serial number, and the run time security key.

Late in the boot procedure, the host unicasts to the card an IP packet containing the run time security key. This packet, termed the “run key packet”, is sent clear channel (unencrypted). This key is the one input by the user — e.g. via the Aculab Configuration Tool (ACT). It overwrites the boot key and is used to prevent any host that does not know the key from controlling the card.

Each security key is a “shared secret” between the host and the Prosody X card, and forms the basis of the security system. It is thus important that the key packets are sent via a secure path.

3.3.3 Using the key

Once the card has been booted, all communications between a host and the card commence with the card issuing a challenge to the host, to which the host must respond. In order to respond correctly, the host must be aware of the run time key (i.e. shared secret), which the card is using. If the host responds incorrectly, the card will terminate the connection.

There is one exception to the above rule. A host can unicast or broadcast a low level protocol packet to the card requesting that the card reset. This is sent clear channel (unencrypted) and is not preceded by a challenge / response procedure. It's purpose is to allow the user to regain control of a card if something has gone wrong — so it's deliberately simple in operation. Bear in mind the assumption that the local switched network is secure and that low-level packets such as this one can only originate from this network.

3.3.4 Security assumptions

The security procedures are based on various assumptions, including those noted here.

3.3.4.1 Security of local switched network

We assume that the local switched network is itself secure. No such assumption is made about any other network. The assumption is made because, if an attacker has control of a host on the local switched network, there are many ways in which a malicious attack can be performed against arbitrary machines on that network. For reference, these include, but are not limited to:

- ARP cache pollution
- ICMP host redirects
- high bandwidth denial of service attacks
- spanning tree protocol attacks
- Content Addressable Memory (CAM) attacks
- correct operation of Ethernet switches

It may be worth considering the use of an Intrusion Detection System on the network.

In particular, we assume that any host that bootstraps and / or controls a Prosody X card is itself secure. The key is stored on the file system within the V6 tree. There is no point in attempting to hide it because, once a host that controls a Prosody X card has been compromised, the card can be compromised because the key is already in use.

3.3.4.2 Handling of denial of service attacks

We assume that the detection and mitigation of denial of service attacks is performed elsewhere. The card's only valid behaviour is to do its best to continue to run. Aculab cannot guarantee correct behaviour during denial of service attacks because both the card itself and the network, which it uses to communicate, are likely to be overloaded and thus performing poorly.

3.3.4.3 No physical access to hardware

We assume that a malicious attacker does not have physical access to the Prosody X card or the host controlling it.

3.3.4.4 Access to run time security key

We assume that the run time security key is kept secret.

3.3.4.5 Access to security mechanism

It is logical to assume that others may have access to the challenge / response mechanism: Aculab's software is freely available to download and, even if it was not, Aculab's customer base would still require access to it. Access to this mechanism is of limited use without the security key. For example, access to the source code of a well-written secure sockets implementation does not in itself make the secure sockets insecure.

3.3.5 Summary

It can be seen that the security key procedures above rely on the run-time security key being kept secret and on the integrity of the switched Ethernet between the host and the card. Packets unicast from the host to the card must arrive only at the card. This should be achieved in normal operation provided no Ethernet hub is in the network path, no Ethernet switch has been compromised and no Ethernet switch is malfunctioning.

3.4 Vulnerabilities

Now that the security mechanisms have been described, we can analyse various types of attack, make an assessment of how vulnerable the Prosody X system (the Prosody X card and host software) is to each and suggest ways to mitigate them.

Bear in mind the assumptions on which the security procedures are based, particularly that the local switched network is secure.

3.4.1 Blind attackers

A blind attacker is defined as someone who has access to generally available attacking tools but is unaware that the host being attacked is a Prosody X card. We further assume that he is on a different switched network from the card.

The security key mechanism in use makes it very difficult for a blind attacker to affect the behaviour of a card by sending it specially crafted packets, scanning its ports, etc. A blind attacker would not be aware of the run time security key or even the challenge / response mechanics.

3.4.2 Accidental or deliberate use of Aculab software by a third party

This scenario ignores the case where, as a deliberate policy, more than one host knows the runtime security key and has been configured to boot the card.

So, assume that there is a malicious host which has (freely available from website) Aculab software installed on it, and which has the system's Prosody X card entered into its Prosody X card list. If the malicious host is configured to boot the card, it must have been added to the list with knowledge of the card serial number. Otherwise, it must have been added with knowledge of either serial number or IP address or both.

3.4.2.1 Malicious host on local switched network

If the malicious host has been configured to boot the card and is on the same switched network as the card, there will be a race condition each time the card boots, with either the correct or the malicious host booting it successfully.

If both hosts use the same run-time security key, then security was compromised at the point when the key was leaked. If they use different keys, only the host which booted the card will be able to control it, meaning that the correct host may not be able to control the card.

3.4.2.2 Second host on different network

If the second host is on a different network from the card then it cannot boot the card, so it makes no difference whether or not this host is configured to do so. The second

host will be able to control the card only if the run-time security key is known.

3.4.3 Sighted attackers

A sighted attacker is defined as someone who has access to generally available attacking tools and is aware that the host being attacked is a Prosody X card. He is therefore aware of which TCP/UDP ports are in use. However, he doesn't have access to the Aculab host software.

3.4.3.1 Attacker on same switched network

If the attacker is on the same switched network as the card, he could prevent a card from booting by sending it a suitably crafted packet in the window between the card broadcasting its first boot-me packet and it receiving a response from the correct host, though this would require either significant experimentation or analysis of the packet stream between a Prosody X card and an Aculab host.

An attacker on this network could reset the card by sending a suitable low-level protocol packet to it. He could also attack arbitrary machines on the network in various ways, see the *Security Assumptions* section above.

Again, we assume that the local switched network is secure.

3.4.3.2 Attacker on same or different switched network

Once the card has booted, the challenge / response procedures restrict his ability to exploit classic vulnerabilities (e.g. port scan based attacks) to a level similar to that of the blind attacker.

A more fruitful avenue would be a man in the middle attack. This refers to a scenario in which a malicious host sits between the card and the correct host, intercepting and sending on all packets sent between them. Packets could either be sent on unchanged, in which case the system will function as desired (subject to any delays and bandwidth restrictions introduced by the malicious host) or changed, in which case the system may function differently.

If a man in the middle attack is in progress at boot time, the malicious host will learn the key. If it is in progress at run time only, it will not. Note that the malicious host does not need to be aware of the key to perform this attack — it merely passes on the challenge / response, and the data is not encrypted. Encrypting the data would make it harder for the malicious host to manipulate the traffic in a controlled way, though it would be able to effectively stall the system.

It would be hard for Aculab software to prevent man in the middle attacks as they are a general network issue — issues such as ARP cache pollution are relevant here.

Note that, in a modification of the man in the middle attack, it would be possible for a malicious host to pretend to be the Prosody X card.

3.4.4 Denial Of Service attack

We define a Denial of Service (DOS) attack as where a malicious host floods the card with packets, aiming to make the system (Prosody X card and host) malfunction or to function slowly or unreliably.

A DOS attack is generally easier to perform if the malicious host is on the same switched network as the card, thus providing it with a high bandwidth connection to the card that is less likely to get detected and throttled than if it was via a router or gateway.

This is very hard for Aculab software to guard against — the best we can do is to keep the card functioning throughout the attack. Application writers may experience delays in functions returning and events being raised. Under extreme conditions, they may also experience Aculab API error return codes. In summary, it's impossible to distinguish between a catastrophic failure of a network or device attached to it, and a network experiencing a harsh DOS attack.

4 Security Features

This section outlines some of the features present in Prosody X devices to facilitate their integration with IP networks.

4.1 Secure host mode

In some configurations, it may be desirable to use the IP telephony features of Prosody X cards without connecting the host system to the IP network on which the Prosody X card operates. For example, when a system is using on-board H.323, all call control and media communication with remote systems is performed by the resources on the card. The host needs only to communicate directly with the card resources.

Secure host mode facilitates such configurations by letting the Prosody X card be configured so that no network traffic can pass between the host Ethernet controller and the external Ethernet connection. The host Ethernet controller will only be able to communicate with the resources on the Prosody X card itself while the resources on the card will be able to communicate freely.

This allows the host to control the card using the host Ethernet controller without being exposed to any traffic from the external network, providing an additional level of protection for the host beyond that offered by running a firewall on the host.

The default configuration is to disable secure host mode. Due to the unusual network configurations that can result from this configuration, it is recommended that it only be used if specifically required. In situations where no external access is required at all, it is recommended that the external Ethernet connection should not be used.

4.1.1 Enabling secure host mode

Secure host mode requires that Prosody IP Firmware 1.0.30 or later be installed on the card.

Secure host mode can be enabled using the `board_cmd` utility:

```
board_cmd <serial number> 0 run touch /config/host_ext_access
```

This change will take effect immediately and will persist until explicitly disabled, even over reboots.

4.1.2 Disabling secure host mode

Secure host mode can be disabled using the `board_cmd` utility:

```
board_cmd <serial number> 0 run rm /config/host_ext_access
```

This change will take effect immediately and will persist until explicitly enabled.

4.1.3 Network configuration considerations

Since secure host mode prevents any access to the external ethernet connection, the host Ethernet controller cannot be configured using DHCP. As usual, the Ethernet controller IP address must be in the same subnet as the IP address of the card resources. However, in order to prevent network contention issues, it is important that the Ethernet controller IP address is not an address assigned elsewhere on the network, for example, an IP address that could be assigned by the DHCP server.

DHCP may still be used to configure the address used by the resources on the card if desired.

In order to allow the card to communicate with the host Ethernet controller, the host Ethernet controller should be assigned an address on the same IP subnet as the card. For example, the following network configuration is valid:

Device	IP address	Netmask
Host Ethernet controller	172.27.133.1	255.255.255.0
Card resources	172.27.133.21	255.255.255.0

Since the host will not be able to use the host Ethernet controller to communicate with a gateway, no gateway should be configured, even if one is present on the network.

For additional robustness, a firewall should be used on the host in addition to secure host mode. This will provide additional protection to the host.

5 Flash upgrades

5.1 Introduction to flash upgrade

5.1.1 What is flash upgrade?

Prosody X cards and media processing chassis are fitted with Power PCs with flash memory containing firmware (flash image), which allows them to boot into a state where they can be configured over a network.

The memory is split into halves with each half containing a flash image. These are termed the main and backup images. A flash upgrade will overwrite the main image with a different version. The backup image cannot be overwritten in the field and is provided so that a card can be made to boot even if an upgrade of the main image went wrong.

5.1.2 Why is it needed?

Aculab may release new versions of the firmware image from time to time, for example to fix bugs or enhance performance. Each version of Aculab host software is tested with a specified version of the firmware image. Therefore there will sometimes be a need to upgrade the firmware image.

5.1.3 When should it be carried out?

Each version of Aculab host software is tested with a specified version of the flash image and also contains the files that comprise this image. Having installed a new version of host software, you should find out if the firmware image is up to date — see section 5.2.1. If it is not up to date then a flash upgrade should be carried out.

5.2 Flashing Prosody X cards

A flash upgrade can be performed on a Prosody X card In the text below, `'card'` is used to describe the particular card you're upgrading.

Flash updates are carried out using the `prosody_ip_card_flash` utility.

5.2.1 How to discover the version of flash image on your card

The flash image version is displayed in the output of `configuration_summary`, obtained by calling `acu_prosody_ip_get_device_info()` on the card. There is also a command line tool `prosody_ip_card_mgr` which can be used. Run;

```
prosody_ip_card_mgr info <serial>
```

where `<serial>` is the serial number of the card. This utility is the preferred way of obtaining the version numbers. This information is also logged in the resource manager trace when the card is started and through `config_summary`.

With reference to the `index` numbers below, a Prosody X card will display information on one devices: the base board (device 0), older products had multiple device numbers but for the current product range this is no longer the case.

The output from this tool for an example Prosody X card is shown below.

```
C:\>prosody_ip_card_mgr info 221946
prosody_ip_card_mgr V6.6.2

Using old format
Card 221946:
Type:Aculab Prosody X card
Version:V3.1
MAC address:00:02:1F:00:C3:4B

Local NIC:p4p2
Configure:Yes
Remote:Yes
IPv4:Dynamic
IPv6:/0
Key:mysitekey
Status:In service

2 ethernet ports:
0: Connected at 1000Mbit full duplex and active
1: Disconnected

Device 0:
Model:AC5700 Prosody X
Serial number:221946
Version:3.1
Bootloader:U-Boot 2010.03 1.2.3 (Prosody-X Rev3.x Oct 17 2011 - 12:12:
Firmware:Aculab Prosody IP Firmware 1.0.142.314
3.3V:3.394V
12V:12.281V
Temperature 0:39C
Temperature 1:29C
```

5.2.2 Flashing the card using prosody_ip_card_flash

The `prosody_ip_card_flash` utility is used to upgrade the flash images on the card under the control of the host. From a command line prompt run:

```
prosody_ip_card_flash <serial> <index>
```

Where `<serial>` is the serial number of the card and `<index>` is the flash image you're interested in which for current products is always 0:

```
index 0 = the card Power PC
```

This will check the card to see if any firmware updates can be applied to the card.

The output from this tool, when run in this way, for an example Prosody X base card with out of date Kernel and Ramdisk images is shown below.

```
C:\>prosody_ip_card_flash.exe 179770 0
Flash update tool V6.6.2

Checking for updates to 222680, AC5700 Prosody X 3.1
PXV3_DSI_FPGA is up to date
Bootloader is up to date
Ethernet_FPGA is up to date
TDM_CTRL_FPGA is up to date
TDM_FPGA is up to date
Kernel is up to date
Ramdisk is up to date
DTB is up to date
Card firmware is not up to date
```

If any updates are required, re-run the command with the additional `-u` option:

```
prosody_ip_card_flash <serial> <index> -u
```

CAUTION

Once the upgrade is in progress you must not attempt to abort it: doing so may cause the flash upgrade to fail rendering the card uncontactable. Should a flash upgrade fail for any reason, please contact Aculab support for advice.

5.2.3 Diagnosing problems following a flash upgrade

If the card is not contactable within a few minutes after performing a flash upgrade then further action is necessary.

Prosody X PCIe card

The base card has LEDs that can be used to diagnose the status of the base card Power PC.

Please refer to the 'Firmware controlled LEDs' section of the ***Prosody X PCIe R3 card installation guide*** for details.