
AlterPath Console Server User Guide

Version 2.1.0 Release 1

This document contains proprietary information of Cyclades and is not to be disclosed or used except in accordance with applicable contracts or agreements.

©Cyclades Corporation, 2002

AlterPath Console Server Version 2.1.0 Release 1 User Guide

October 2002

Copyright © Cyclades Corporation, 2002

We believe the information in this manual is accurate and reliable. However, we assume no responsibility, financial or otherwise, for any consequences of the use of this product or manual. This manual is published by Cyclades Corporation, which reserves the right to make improvements or changes in the products described in this manual as well as to revise this publication at any time and without notice to any person of such revision or change. The operating system covered in this manual is v 2.1.0. All brand and product names mentioned in this publication are trademarks or registered trademarks of their respective holders.

Cyclades, AlterPath ACS16, and AlterPath ACS32 are registered trademark of Cyclades Corporation.

Microsoft, Windows 95, 98, XP, ME, NT, and 2K are trademarks of Microsoft Corporation.

UNIX is a trademark of UNIX System Laboratories, Inc.

Linux is a registered trademark of Linus Torvald.

All rights reserved. This document may not, in whole or part, be copied, photocopied, reproduced, translated, or converted to any electronic or machine-readable form without the prior written consent of Cyclades Corporation, 41829 Albrae Street, Fremont, CA 94538, USA. Telephone (510) 770-9727. Fax (510) 770-0355. www.cyclades.com.

Software Release 2.1.0

Document Revision Number 2.1.0-Draft 10.1

Table of Contents

Preface

Purpose	11
Audience and User Levels	11
New Users	11
Power Users	11
How to use this Guide	12
Additional Documentation and Help	13
Conventions and Symbols	13
Fonts	13
Hypertext Links	13
Glossary Entries	13
Note Box Icons	14
Quick Steps	15

Chapter 1 - Introduction and Overview

Introducing Cyclades	17
The AlterPath Console Server	17
Console Access Server	18
What's in the box	20
Safety Instructions	22
Replacing the Battery	23
FCC Warning Statement	24

Chapter 2 - Installation and Configuration

Introduction	26
System Requirements	26
Default Configuration Parameters	27
Pre-Install Checklist	28
Task List	28
The Wizard	29
Quick Start	30
Configuration using a Console	31
Configuration using a Web browser	33
Configuration using Telnet	39
The Installation and Configuration Process	42
Task 1: Connect the AlterPath Console Server to the Network and other Devices	42

Table of Contents

Task 2: Configure the COM Port Connection and Log In	45
Task 3: Modify the System Files	47
Task 4: Edit the pslave.conf file	50
Task 5: Activate the changes	53
Task 6: Test the configuration	53
Task 7: Save the changes	54
Task 8: Reboot the AlterPath Console Server	54

Chapter 3 - Additional Features

Introduction	55
Configuration Wizard - Basic Wizard	56
Using the Wizard through your Browser	60
Access Method	61
Parameters Involved and Passed Values	62
Configuration for CAS	63
Configuration for TS	74
Configuration for Dial-in Access	81
Authentication	84
Parameters Involved and Passed Values	84
Configuration for CAS	86
Configuration for TS	93
Configuration for Dial-in Access	93
Clustering	93
Parameters Involved and Passed Values	94
Centralized Management - The Include File	98
CronD	102
Parameters Involved and Passed Values	102
Configuration for CAS	102
Configuration for TS	104
Configuration for Dial-in Access	105
Data Buffering	105
Introduction	105
Linear vs. Circular Buffering	106
Parameters Involved and Passed Values	106
Configuration for CAS	108
DHCP	116
Parameter Involved and Passed Values	116
Configuration for CAS	118
Configuration for TS	118

Table of Contents

Configuration for Dial-in Access	118
Dual Power Management	119
Parameters Involved and Passed Values	119
Configuration for CAS	119
Configuration for TS	119
Configuration for Dial-in Access	120
Filters	120
Parameters Involved and Passed Values	120
Configuration for CAS	124
Configuration for TS	126
Configuration for Dial-in Access	126
Generating Alarms	126
Port Slave Parameters Involved with Generating Alarms	127
vi Method	127
Browser Method	127
Wizard Method	128
Syslog-ng Configuration to use with Alarm Feature	131
Alarm, Sendmail, Sendsms and Snpmptrap	133
Help	140
Help Wizard Information	140
Help Command Line Interface Information	141
NTP	143
Parameters Involved and Passed Values	143
Configuration for CAS	144
Configuration for TS	145
Configuration for Dial-in Access	145
PCMCIA	146
Supported Cards	146
Tools for Configuring and Monitoring PCMCIA Devices	146
Ejecting Cards	147
PCMCIA Network Configuration	147
Modem PC Cards	149
Wireless LAN PC Cards	149
Ports Configured for Dial-in Access	151
Ports Configured as Terminal Servers	153
TS Setup Scenario	154
TS Setup Wizard	155
Serial Settings	158
Parameters Involved and Passed Values	158
Configuration for CAS	160

Table of Contents

Configuration for TS	167
Configuration for Dial-in Access	170
Session Sniffing	170
Versions 1.3.2 and earlier	170
Versions 1.3.3 and later	171
Parameters Involved and Passed Values.	173
Configuration for CAS	174
Wizard Method	175
SNMP	180
Configuration for CAS	181
Configuration for TS	182
Configuration for Dial-in Access	182
Syslog	182
Port Slave Parameters Involved with syslog-ng	183
Configuration for CAS	183
Configuration for TS	187
Configuration for Dial-in Access	188
The Syslog Functions	188
Terminal Appearance	202
Parameters Involved and Passed Values.	202
Browser Method	202
Wizard Method	203
Time Zone.	207
How to set Date and Time	208

Appendix A - New User Background Information

Users and Passwords.	210
Linux File Structure	210
Basic File Manipulation Commands	211
The vi Editor	212
The Routing Table	214
Secure Shell Session	215
The Process Table.	219
TS Menu Script	220

Table of Contents

Appendix B - Cabling, Hardware, and Electrical Specifications

General Hardware Specifications	223
The RS-232 Standard	224
Cable Length	226
Connectors.	226
Straight-Through vs. Crossover Cables	227
Which cable should be used?.	228
Cable Diagrams	229

Appendix C - The pslave Configuration File

Introduction	235
Configuration Parameters	235
Additional AlterPath Console Server Options for a CAS	235
TS Parameters	251
Dial-in Access Parameters	252

Appendix D - Linux-PAM

Introduction	255
The Linux-PAM Configuration File	257
Configuration File Syntax	257
Newest Syntax	260
Module Path	261
PAM Kerberos	263
PAM LDAP	263
Arguments	264
Directory-based Configuration	265
Default Policy	266
Reference	272

Appendix E - Customization and the Cyclades Developer Kit

Introduction	273
The Customization Process	274
The Cyclades Development Kit	274

Table of Contents

Appendix F - Software Upgrades and Troubleshooting

Upgrades	275
The Upgrade Process	275
Troubleshooting	277
Flash Memory Loss	277
Hardware Test	280
Port Test	280
Port Conversation	281
Test Signals Manually	281
Single User Mode	282
Troubleshooting the Web Configuration Manager	284
What to do when the initial Web page does not appear	284
How to restore the Default Configuration of the Web Configuration Manager	284
Using a different speed for the Serial Console	284
How to connect to serial ports from the browser	285
CPU LED	290

Appendix G - Certificate for HTTP Security

Introduction	292
Procedure	292

Appendix H - IPSEC

Introduction	295
IPsec, Security for the Internet Protocol	295
Applications of IPsec	295
Configuration	296
Before you start	296
Set up and test networking	296
Enabling IPsec	297
Quick Start	297
"Road Warrior" remote access	297
ACS-to-network VPN	299
Setting up RSA authentication keys	301
Generating an RSA key pair	301
Exchanging authentication keys	301
The Configuration File	302

Table of Contents

Description	302
Conn Sections	304
Config Sections	309
Recommended Configuration	311
IPsec Usage	312
The IPsec Daemon	312
Adding and Removing a Connection	312
Starting and Stopping a Connection	313

List of Wiz Application Parameters

Basic Parameters (wiz)	314
Authentication Parameters (wiz --auth)	314
Terminal Appearance Parameters (wiz --tl)	315
Alarm Parameter (wiz --al)	315
Data Buffering Parameters (wiz --db)	315
Sniffing Parameters (wiz --snf)	316
Syslog Parameters (wiz --sl)	316
Terminal Server Profile Other Parameters (wiz --tso)	316
Access Method Parameters (wiz --ac <type>)	317
Serial Settings Parameters (wiz --sset <type>)	318

List of Figures	319
---------------------------	-----

List of Tables.	321
-------------------------	-----

Glossary.	323
-------------------	-----

Index	327
-----------------	-----

Table of Contents

This page has been left intentionally blank.

Preface

Purpose

The purpose of this guide is to provide instruction for users to independently install, configure, and maintain the AlterPath Console Server. This manual should be read in the order written, with exceptions given in the text. *Whether or not you are a UNIX user, we strongly recommend that you follow the steps given in this manual.*

Audience and User Levels

This guide is intended for the user who is responsible for the deployment and day-to-day operation and maintenance of the AlterPath Console Server. It assumes that the reader understands networking basics and is familiar with the terms and concepts used in Local and Wide Area Networking. UNIX and Linux users will find the configuration process very familiar. It is not necessary to be a UNIX expert, however, to get the AlterPath Console Server up and running. There are two audiences or user levels for this manual:

New Users

These are users new to Linux and/or UNIX with a primarily PC/Microsoft background. You might want to brush up on such things as common Linux/UNIX commands and how to use the vi editor prior to attempting installation and configuration. This essential background information appears in [Appendix A - New User Background Information](#). It is recommended that New Users configure the AlterPath Console Server using a Web browser, however, New Users can also configure the AlterPath Console Server with vi, the Wizard or the Command Line Interface (CLI).

Power Users

These are UNIX/Linux experts who will use this manual mostly for reference. Power Users can choose between configuring the AlterPath Console Server via Web browser, vi, Wizard, or CLI.

Each configuration task will be separated into a section (a clickable link on the PDF file) for each user type. Users then can skip to the appropriate level that matches their expertise and comfort level.

Preface

How to use this Guide

This guide is organized into the following sections:

- [Chapter 1 - Introduction and Overview](#) contains an explanation of the product and its default CAS setup. It also includes safety guidelines to be followed.
- [Chapter 2 - Installation and Configuration](#) explains how the AlterPath Console Server should be connected and what each cable is used for. It describes the basic configuration process to get the AlterPath Console Server up and running for its most common uses.
- [Chapter 3 - Additional Features](#) is dedicated to users wanting to explore all available features of the AlterPath Console Server. It provides configuration instructions for syslog, data buffers, authentication, filters, DHCP, NTP, SNMP, clustering, and sniffing.
- [Appendix A - New User Background Information](#) contains information for those who are new to Linux/UNIX.
- [Appendix B - Cabling, Hardware, and Electrical Specifications](#) has detailed information and pinout diagrams for cables used with the AlterPath Console Server.
- [Appendix C - The pslave Configuration File](#) contains example files for the various configurations as well as the master file.
- [Appendix D - Linux-PAM](#) enables the local system administrator to choose how to authenticate users.
- [Appendix E - Customization and the Cyclades Developer Kit](#) provides instruction for those who wish to create their own applications.
- [Appendix F - Software Upgrades and Troubleshooting](#) includes solutions and test procedures for typical problems. In addition, instruction is provided for connection to serial ports through the browser to view the server screen.
- [Appendix G - Certificate for HTTP Security](#) provides configuration information that will enable you to obtain a Signed Digital Certificate.
- [Appendix H - IPSEC](#) provides encryption and authentication services at the IP (Internet Protocol) level of the network protocol stack.
- The [Glossary](#) provides definitions for commonly-used terms in this manual.

Preface

Additional Documentation and Help

There are other Cyclades documents that contain background information about Console Port Management and the Cyclades product line. These are:

- The Cyclades Console Port Management Guide
- The Cyclades Product Catalog

For the most updated version of Cyclades' documentation, use the following Web address:

<http://www.cyclades.com/support/downloads.php>

Conventions and Symbols

This section explains the significance of each of the various fonts, formatting, and icons that appear throughout this guide.

Fonts

This guide uses a regular text font for most of the body text and `Courier` for data that you would input, such as a command line instruction, or data that you would receive back, such as an error message. An example of this would be:

```
telnet 200.200.200.1 7001
```

Hypertext Links

References to another section of this manual are hypertext links that are [underlined](#) (and are also blue in the PDF version of the manual). When you click on them in the PDF version of the manual, you will be taken to that section.

Glossary Entries

Terms that can be found in the glossary are underlined and slightly larger than the rest of the text. These terms have a hypertext link to the glossary.

Preface

Note Box Icons

Note boxes contain instructional or cautionary information that the reader especially needs to bear in mind. There are five levels of note box icons:



Tip. An informational tip or tool that explains and/or expedites the use of the AlterPath Console Server.



Important! An important tip that should be read. Review all of these notes for critical information.



Warning! A very important type of tip or warning. Do not ignore this information.



DANGER! Indicates a direct danger which, if not avoided, may result in personal injury or damage to the system.



Security Issue. Indicates security-related information where it is relevant.

Preface

Quick Steps

Step-by-step instructions for installing and configuring the AlterPath Console Server are numbered with a summarized description of the step for quick reference. Underneath the quick step is a more detailed description. Steps are numbered 1, 2, 3, etc. Additionally, if there are sub-steps to a step, they are indicated as Step A, B, C, and are nested within the Step 1, 2, 3, etc. For example:

Step 1: Modify files.

You will modify four Linux files to let the AlterPath Console Server know about its local environment.

Step A: Modify `pslave.conf`.

Open the file `pslave.conf` and add the following lines . . .

Preface

This page has been left intentionally blank.

Introduction and Overview

Introducing Cyclades

Cyclades is a data center fault management company that enables remote management of servers, network equipment and automation devices. Its products help data center managers at enterprise, telecommunication and Internet companies to maximize network and server availability. This results in decreased maintenance costs, increased efficiency and productivity, along with greater control, freedom and peace of mind. Cyclades' advantage is providing scalable products leveraging Linux technology for flexibility and ease of customization.

The AlterPath Console Server

The AlterPath Console Server is line of Console Access and Terminal Servers that allow both local and dial-in access for in-band and out-of-band network management. They run an embedded version of the Linux operating system. Configuration of the equipment is done by editing a few plain-text files, and then updating the versions of the files on the AlterPath Console Server. The files can be edited using the vi editor provided or on another computer with the environment and text editor of your choice. The default “box profile” of the product is that of a Console Access Server.

You can access the AlterPath Console Server via three methods:

- A console directly connected to the AlterPath Console Server
- Telnet/ssh over a network
- A browser

And configure it with any of the following four options:

- vi
- Wizard
- Browser
- Command Line Interface (CLI) - only for certain configuration parameters.

Introduction and Overview

The AlterPath ACS comes with a dual power supply and two PCMCIA slots. The AlterPath CS has a single power supply and no PCMCIA slot. They are similar in every other way. With the AlterPath Console Server set up as a Console Access Server, you can access a server connected to the (A)CS through the server's serial console port from a workstation on the LAN or WAN. There is no authentication by default, but the system can be configured for authentication to be performed by a Radius server, a TacacsPlus server, or even by a local database. Either telnet or ssh (a secure shell session) can be used. See [Appendix A - New User Background Information](#) for more information about ssh. The instructions in [Chapter 2 - Installation and Configuration](#) will set up a default, fully-functional CAS environment. More options can be added after the initial setup, as illustrated in [Chapter 3 - Additional Features](#).

Console Access Server

An example of a CAS environment is shown in [Figure 1: Console Access Server diagram](#). This configuration example has local authentication, an Ethernet interface provided by a router, and serially-connected workstations.

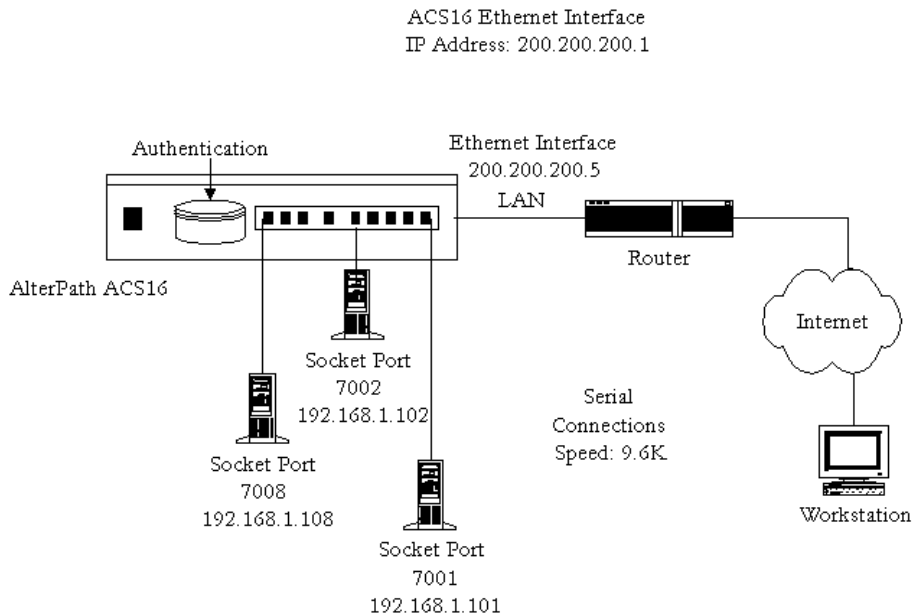


Figure 1: Console Access Server diagram

Introduction and Overview

The following diagram, [Figure 2: CAS diagram with various authentication methods](#), shows additional scenarios for the AlterPath Console Server: both remote and local authentication, data buffering, and remote access.

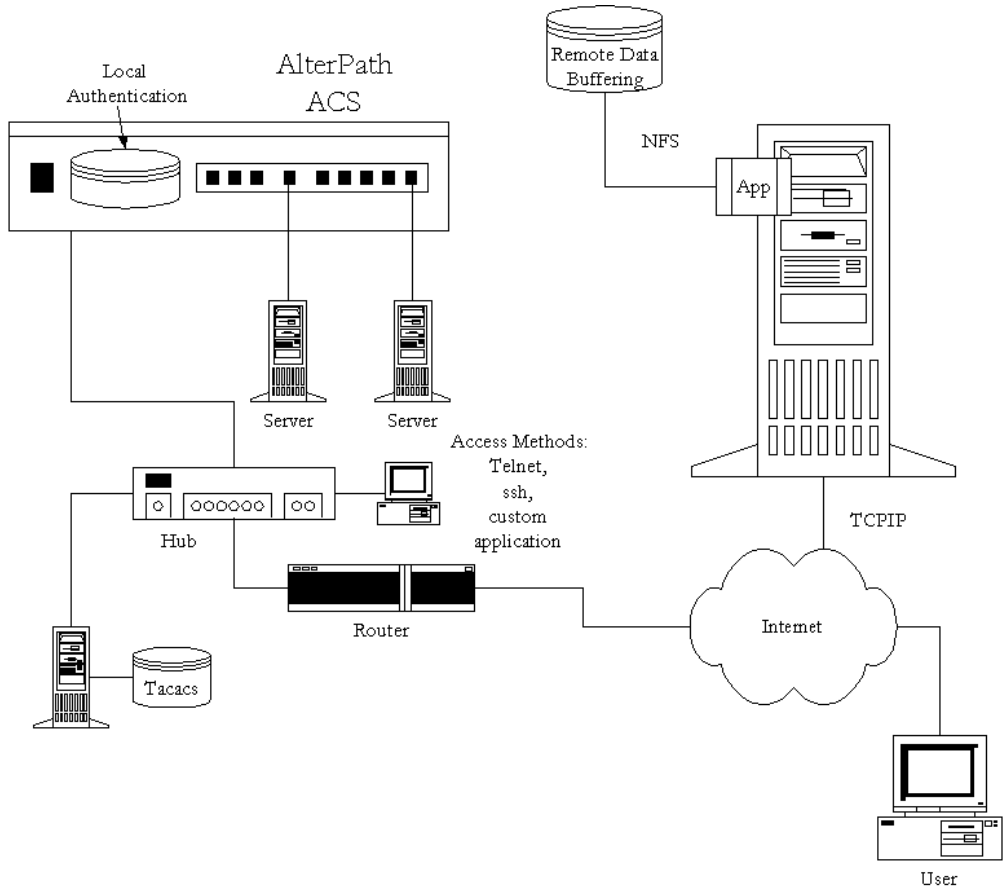


Figure 2: CAS diagram with various authentication methods

Introduction and Overview

What's in the box

There are several models of the AlterPath Console Server with differing numbers of serial ports. The following figures show the main units and accessories included in each package. The RJ-45 straight-through cable is the main cable that you will use. After configuration, it can be used with the same adapter to connect to the server. Four adapters are included: two RJ-45 to DB-9 (male and female) and two RJ-45 to DB-25 (male and female). Select the adapter appropriate to your COM port. Two power cables, a modem cable, manual and mounting kit are also included in the box. The Sun Netra adapter, a Cat.5e Inline Coupler, also attaches to the Console Cable. The loop-back connector is provided for convenience in case hardware tests are necessary.

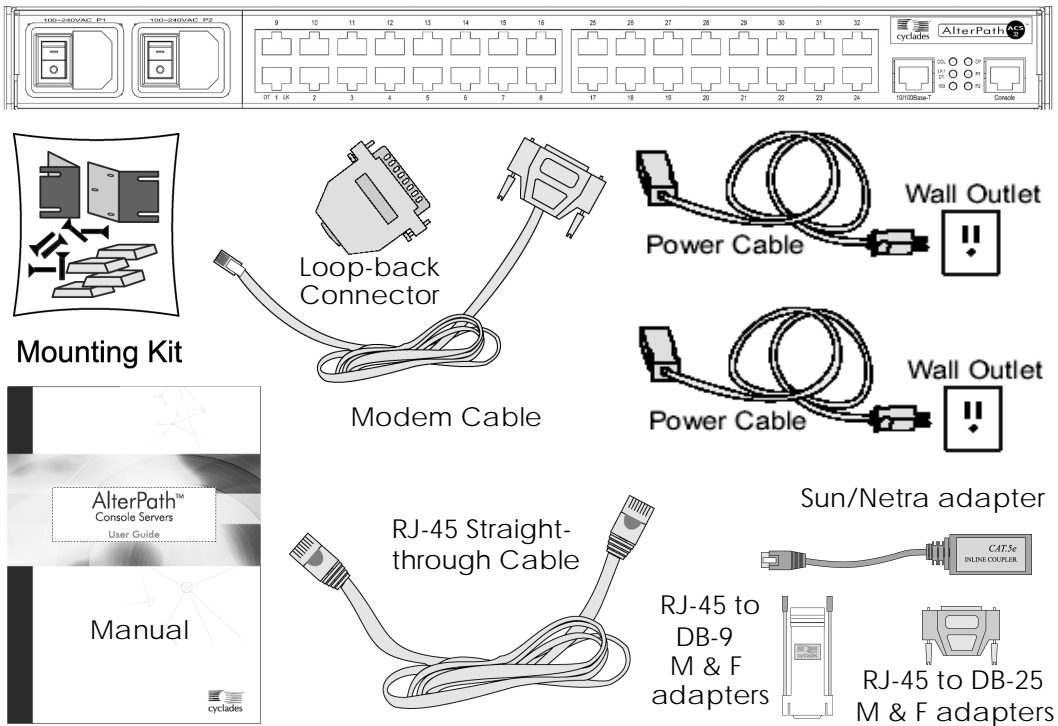


Figure 3: The AlterPath ACS32, its cables, connectors and other box contents

Introduction and Overview

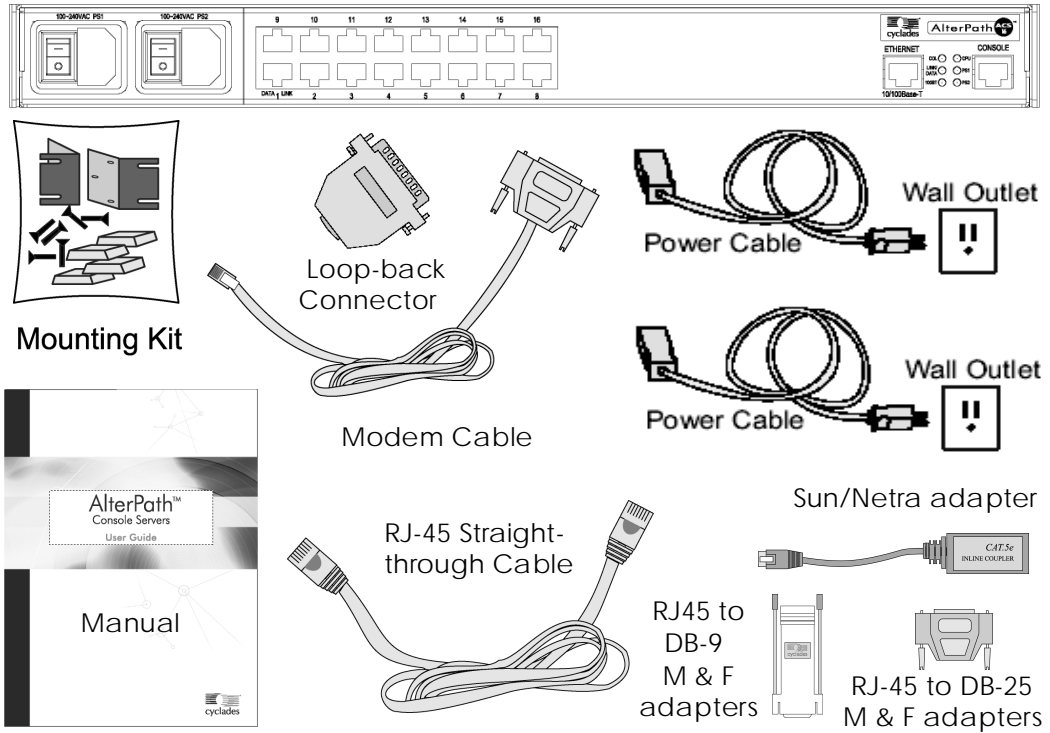


Figure 4: The AlterPath ACS16, its cables, connectors and other box contents

Introduction and Overview

Safety Instructions

Read all the following safety guidelines to protect yourself and your AlterPath Console Server.



DANGER! Do not operate your AlterPath Console Server with the cover removed.



DANGER! In order to avoid shorting out your AlterPath Console Server when disconnecting the network cable, first unplug the cable from the equipment and then from the network jack. When reconnecting a network cable to the equipment, first plug the cable into the network jack, and then into the equipment.



DANGER! To help prevent electric shock, plug the AlterPath Console Server into a properly grounded power source. The cable is equipped with a three-prong plug to help ensure proper grounding. Do not use adapter plugs or remove the grounding prong from the cable. If you have to use an extension cable, use a three-wire cable with properly grounded plugs.



Important! To help protect the AlterPath Console Server from electrical power fluctuations, use a surge suppressor, line conditioner, or uninterruptible power supply.



Important! Be sure that nothing rests on the cables of the AlterPath Console Server and that they are not located where they can be stepped on or tripped over.

Introduction and Overview



Important! Do not spill food or liquids on the AlterPath Console Server. If it gets wet, contact Cyclades.



DANGER! Do not push any objects through the openings of the AlterPath Console Server. Doing so can cause fire or electric shock by shorting out interior components.



Important! Keep your AlterPath Console Server away from heat sources and do not block cooling vents.

Working inside the AlterPath Console Server

Do not attempt to service the AlterPath Console Server yourself, except when following instructions from Cyclades Technical Support personnel. In the latter case, first take the following precautions:

- Turn the AlterPath Console Server off.
- Ground yourself by touching an unpainted metal surface on the back of the equipment before touching anything inside it.

Replacing the Battery

A coin-cell battery maintains date and time information. If you have to repeatedly reset time and date information after turning on your AlterPath Console Server, replace the battery.

Introduction and Overview



DANGER! A new battery can explode if it is incorrectly installed. Replace the 3-Volt CR2032 battery only with the same or equivalent type recommended by the battery manufacturer. Discard used batteries according to the battery manufacturer's instructions.

FCC Warning Statement

The AlterPath Console Server has been tested and found to comply with the limits for Class A digital devices, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the Installation & Service Manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user is required to correct the problem at his or her own expense.

Notice about FCC compliance for the AlterPath ACS16 and the AlterPath ACS32

In order to comply with FCC standards the AlterPath Console Server require the use of a shielded CAT 5 cable for the Ethernet interface. Notice that this cable is not supplied with either of the products and must be provided by the customer.

Canadian DOC Notice

The *AlterPath Console Server* does not exceed the Class A limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

L'AlterPath Console Server n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le règlement sur le brouillage radioélectrique édicté par le Ministère des Communications du Canada.

Introduction and Overview

This page has been left intentionally blank.

Chapter 2 - Installation and Configuration

Introduction

This chapter will allow you to install and configure AlterPath Console Server as the default CAS configuration. *Please read the entire chapter before beginning.* A basic installation and configuration should take a half hour at the most, either done manually or with the Wizard.

The AlterPath Console Server operating system is embedded Linux. If you are fairly new to Linux, you will want to brush up prior to proceeding with this chapter with the essential background information presented in [Appendix A - New User Background Information](#). *Even if you are a UNIX user and find the tools and files familiar, do not configure this product as you would a regular Linux server.*

The chapter is divided into the following sections:

- [System Requirements](#)
- [Default Configuration Parameters](#)
- [Pre-Install Checklist](#)
- [Task List](#)
- [The Wizard](#)
- [Quick Start](#)
- [The Installation and Configuration Process](#)

System Requirements

Cyclades recommends either of the following specifications for a configuration of the AlterPath Console Server:

- A workstation with a console serial port, or
- A workstation with Ethernet and TCP/IP topology

Chapter 2 - Installation and Configuration

The following table shows the different hardware required for various configuration methods:

Table 1: Hardware vs. Configuration Methods

Hardware	Configuration Method
Console, Console Cable (constructed from RJ-45 straight through cable + adapter)	vi, Wizard, or CLI
Workstation, Hub, Ethernet Cables	vi, Wizard, CLI, or browser

If you will be using vi, the files that need to be changed are discussed in [Configuration using Telnet](#) in this chapter. If you will be using the Wizard, basic Wizard access can be found under [Configuration Wizard - Basic Wizard](#) in [Chapter 3 - Additional Features](#) and specifics of this method are discussed under the appropriate option title in the same chapter. If you choose the browser method, the [Quick Start](#) in this chapter shows the screen flow and input values needed for this configuration mode. If you choose the CLI (Command Line Interface) method, this allows you to configure certain parameters for a specified serial port or some network-related parameters. Specifics of this method is discussed under the appropriate option title in [Chapter 3 - Additional Features](#).

Default Configuration Parameters

- Ethernet 192.168.160.10
- Netmask 255.255.255.0
- CAS configuration
- socket_server in all ports (access method is telnet)
- 9600 bps, 8N1
- No Authentication

Chapter 2 - Installation and Configuration

Pre-Install Checklist

There are several things you will need to confirm prior to installing and configuring the AlterPath Console Server:

<i>Root Access</i>	You will need Root Access on your local UNIX machine in order to use the serial port.
<i>HyperTerminal, Kermit, or Minicom</i>	If you are using a PC, you will need to ensure that HyperTerminal is set up on your Windows operating system. If you have a UNIX operating system, you will be using Kermit or Minicom.
<i>IP Address of PC or terminal, AlterPath Console Server, NameServer, and Gateway</i>	You will need to locate the IP address of your PC or workstation, the AlterPath Console Server, and the machine that resolves names on your network. Your Network Administrator can supply you with these. If there is outside access to the LAN that the AlterPath Console Server will be connected with, you will need the gateway IP address as well.
<i>Network Access</i>	You will need to have a NIC card installed in your PC to provide an Ethernet port, and have network access.

Task List

There are eight key tasks that you will need to perform to install and configure the AlterPath Console Server:

[Task 1: Connect the AlterPath Console Server to the Network and other Devices.](#)

[Task 2: Configure the COM Port Connection and Log In.](#)

[Task 3: Modify the System Files.](#)

[Task 4: Edit the pslave.conf file.](#)

[Task 5: Activate the changes.](#)

[Task 6: Test the configuration.](#)

[Task 7: Save the changes.](#)

Chapter 2 - Installation and Configuration

[Task 8: Reboot the AlterPath Console Server.](#)

The Wizard

The eight key tasks can also be done through a wizard in the 2.1 plus versions of the AlterPath Console Server.

Basic Wizard

The Basic Wizard will configure the following parameters:

- IP Address
- Netmask
- Default Gateway
- DNS Server
- Domain

Basic Wizard access is covered in the Quick Start in this chapter and also in [Configuration Wizard - Basic Wizard](#) in [Chapter 3 - Additional Features](#).

Custom Wizard

Further configuration of the AlterPath Console Server can be done through one of several customized wizards. These procedures are explained under their respective topic heading in [Chapter 3 - Additional Features](#). There are custom wizards for the following optional configurations:

- [Access Method](#)
- [Generating Alarms](#)
- [Authentication](#)
- [Data Buffering](#)
- [Help](#)
- [Serial Settings](#)

Chapter 2 - Installation and Configuration

- [Session Sniffing](#)
- [Syslog](#)
- [Terminal Appearance](#)
- [TS Setup Wizard](#) (These are additional configuration parameters applied only to the TS profile.)

Quick Start

This Quick Start gives you all the necessary information to quickly configure and start using the AlterPath Console Server as a Console Access Server (CAS). The complete version of this process is listed later in this chapter under [The Installation and Configuration Process](#). New Users may wish to follow the latter instruction set, as this Quick Start does not contain a lot of assumed knowledge, and could be confusing to the New User.

You can configure the AlterPath Console Server by any one of four methods:

- Console
- Browser
- Telnet
- CLI (Command Line Interface)

If you have a serial port that you can use as a console port, use the Console method. If you have access to telnet, you can use this method, while [New Users](#) may prefer the Browser method for its user-friendliness.



Important! Take care when changing the IP address of the AlterPath Console Server. Confirm the address you are changing it to. (You may want to write it down.)

Chapter 2 - Installation and Configuration

Configuration using a Console

Step 1: Connect the console cable.

Connect the console cable (created from the RJ-45 straight through cable and the appropriate adapter) to the port labeled “Console” on the AlterPath Console Server with the RJ-45 connector end, and to your PC’s available COM port with the adapter’s serial port end.

Step 2: Power on the AlterPath Console Server.

After the AlterPath Console Server finishes booting, you will see a login prompt on the console screen.

Step 3: Enter root as login name and tslinux as password.

Step 4: Type wiz and press Enter.

A wizard configuration screen will appear, asking you a series of questions.

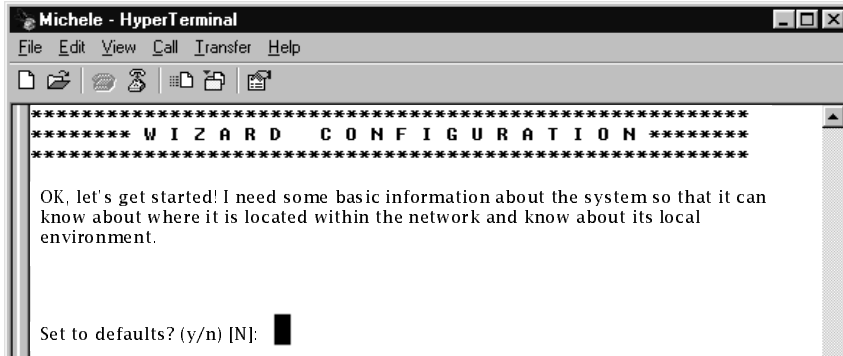


Figure 5: The initial wizard configuration screen

You will want to configure the following settings:

- Hostname
- System IP
- Domain Name

Chapter 2 - Installation and Configuration

- Primary DNS Server
- Gateway IP
- Network Mask

After you input the requested parameters you will receive a confirmation screen:

Your current configuration parameters are:

Hostname : CAS

System IP : 192.168.160.10

Domain name : cyclades.com

Primary DNS Server : 197.168.160.200

Gateway : 192.168.160.10

Network Mask : 255.255.255.0

If the parameters are correct, “Y” should be typed; otherwise, type “N” and then “C” when asked to change the parameters or quit the program. After the parameters are confirmed, the next question will be whether to save the configuration to flash. Select “Y” to make the new configuration permanent in non-volatile memory.

After you confirm and save the basic parameters, you will be presented with the shell prompt. From there, either select to continue configuration using the vi editor or continue using a browser.

The AlterPath Console Server is now configured as a CAS with its new IP address, with no authentication, and accepting telnet to the serial ports. You can telnet the CAS IP + serial port 1 with the following command:

```
telnet <IP you assigned> 7001
```



Note. Serial port 1 is configured as 9600, 8N1 by default. The server connected to this serial port has to have the same configuration for its serial port.

Chapter 2 - Installation and Configuration

To explore the AlterPath Console Server features, either continue configuration using the vi editor from the console, use a browser from a workstation and point to the AlterPath Console Server, or use the CLI, if appropriate (as indicated in the relevant sections in Chapter 3).

Configuration using a Web browser

The AlterPath Console Server box comes with an IP address pre-configured on its Ethernet interface (192.168.160.10). To access that box using your browser:

Step 1: Connect Hub to workstation and (A)CS.

Your workstation and your (A)CS must be on the same physical LAN. Connect one RJ-45 cable from the Ethernet port of the (A)CS to a spare port from a hub, and another cable from another spare port of that same hub to the workstation used to manage the servers.

Step 2: Add route pointing to the (A)CS IP.

From the workstation, issue a command to add a route pointing to the network IP address of the (A)CS (192.168.160.0) accessed through the workstation's Ethernet interface.

For Linux, the command would be:

```
route add -net 192.168.160.0/24 gw <IP address assigned to the workstation's Ethernet interface>
```

Example: if the workstation has IP address 200.246.93.150 the command would be:

```
route add -net 192.168.160.0/24 gw 200.246.93.150
```

For Windows, the command would be:

```
route add 192.168.160.0 mask 255.255.255.0 <IP address assigned to the workstation's Ethernet interface>
```

Example: if the workstation has IP address 200.246.93.150 the command would be:

```
route add 192.168.160.0 mask 255.255.255.0 200.246.93.150
```

Step 3: Point your browser to 192.168.160.10.

The login page shown in the following figure will appear.

Chapter 2 - Installation and Configuration

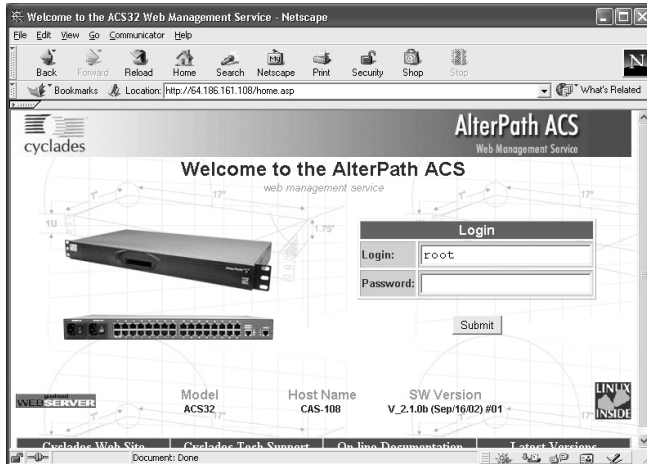


Figure 6: Login page of Web Configuration Manager

Step 4: Enter root as login name and tlinux as password.

Step 5: Click the Submit button.

This will take you to the Configuration & Administration Menu page, shown below.

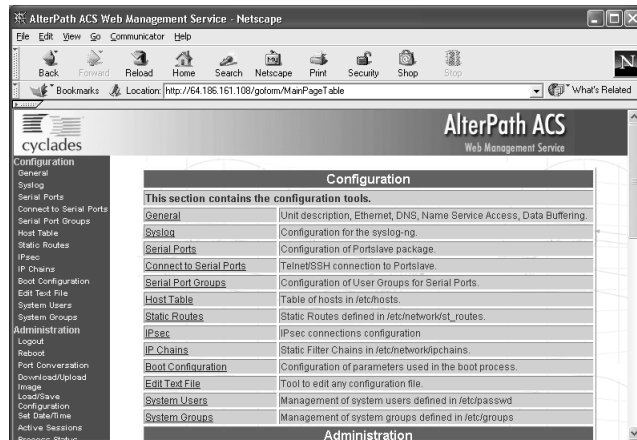


Figure 7: Configuration & Administration Menu page

Chapter 2 - Installation and Configuration

This page gives a brief description of all menu options and allows you to change your password.



Security Issue. Change the root password as soon as possible. The user database for the Web Configuration Manager is different than the system user database, so the root password can be different. See [How to change the Password \(Web Method\)](#).

Step 6: Configure using the General page.

The General page of the Web Configuration Manager is shown in the following figure.

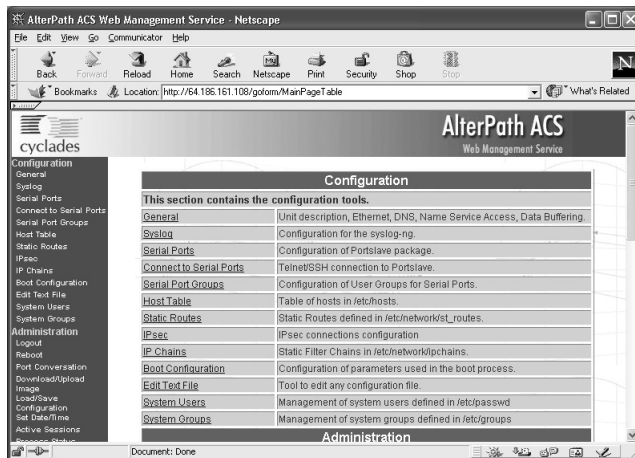


Figure 8: General page

Step 7: Configure parameters presented in the fields.

A menu of links is provided along the left side of the page. A summary of what each link leads to is shown on [Table 2: Configuration Section](#) through [Table 5: Information Section](#).

Step 8: Click on the link Web User Management > Load/Save Configuration.

Step 9: Click the Save Configuration button.

Chapter 2 - Installation and Configuration

Step 10: Click on the link **Administration > Load/Save Configuration**.

Step 11: Click the **Save Configuration to Flash** button.

The configuration was saved in flash but it is not yet running.

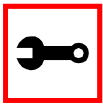
Step 12: Click on **Administration > Restart Processes > Stop cy_ras**.

Step 13: Click on **Start cy_ras**.

If you changed your ethernet IP, you will lose your connection. You will need to use your browser to connect to the new IP.

The new configuration will be valid and running. The AlterPath Console Server is now configured as a CAS with its new IP address, with no authentication, and accepting telnet to the serial ports. You can telnet the CAS IP + serial port 1 with the following command:

```
telnet <IP you assigned> 7001
```



Note. Serial port 1 is configured as 9600, 8N1 by default. The server connected to this serial port has to have the same configuration for its serial port.

To explore the AlterPath Console Server features, either continue configuration using your browser, use the vi editor from the console, or use CLI, if appropriate.

How to change the Password (Web Method)

Step 1: Click on the link **Web User Management > Users**.

Step 2: Select the user *root*.

Step 3: Click on the **Change Password** button.

Step 4: Type the new password twice.

Step 5: Submit the request.

The next page will require a new login.

Chapter 2 - Installation and Configuration

Step 6: Type root and the new password.

Table 2: Configuration Section

Link Name	Description of Page Contents
<i>General</i>	Description, Ethernet, DNS, Name Service Access, Data Buffering
<i>Syslog</i>	Configuration for the syslog-ng
<i>Serial Ports</i>	Configuration for the Portslave package
<i>Connect to Serial Ports</i>	Telnet/SSH connection to Portslave. (See Note Box below.)
<i>Serial Port Groups</i>	User Groups in Serial Ports Configuration
<i>Host Table</i>	Table of hosts in /etc/hosts
<i>Static Routes</i>	Static routes defined in /etc/network/st_routes
<i>IP Tables Filter</i>	Static Firewall IPTables Filter in /etc/network/firewall
<i>Boot Configuration</i>	Configuration of parameters used in the boot process
<i>Edit Text File</i>	Tool to read and edit a configuration file
<i>System Users</i>	Management of system users defined in /etc/password
<i>System Groups</i>	Management of system groups defined in /etc/groups



Note: The link **Connect to Serial Ports** is only available for the ACS16, ACS32, and ACS48. See [“How to connect to serial ports from the browser” on page 285](#).

Chapter 2 - Installation and Configuration

Table 3: Web User Management Section

Link Name	Description of Page Contents
<i>Users</i>	List of users allowed to access the Web server
<i>Groups</i>	List of possible access groups
<i>Access Limits</i>	List of access limits for specific URLs
<i>Load/Save Configuration</i>	Load/Save Web user configuration in /etc/websum.conf

Table 4: Administration Section

Link Name	Description of Page Contents
<i>Logout</i>	Exits the Web Manager
<i>Reboot</i>	Resets the equipment
<i>Send Message</i>	Sends messages to users logged into a serial port
<i>Port Conversation</i>	Initiates a port conversation through a serial port
<i>Download/Upload Image</i>	Uses an FTP server to load and save a kernel image
<i>Load/Save Configuration</i>	Uses flash memory or an FTP server to load or save the (A)CS's configuration
<i>Set Date/Time</i>	Set the (A)CS's date and time
<i>Active Sessions</i>	Shows the active sessions and allows the administrator to kill them
<i>Process Status</i>	Shows the running processes and allows the administrator to kill them
<i>Restart Processes</i>	Allows the administrator to start or stop some processes

Chapter 2 - Installation and Configuration

Table 5: Information Section

Link Name	Description of Page Contents
<i>Interface Statistics</i>	Shows statistics for all active interfaces
<i>DHCP client</i>	Shows the DHCP client information
<i>Serial Ports</i>	Shows the status of all serial ports
<i>Routing Table</i>	Shows the routing table and allows the administrator to add or delete routes
<i>ARP Cache</i>	Shows the ARP cache
<i>IP Chains</i>	Shows IP Chains entries
<i>IP Rules</i>	Shows Firewall, NAT, and IP Accounting rules
<i>IP Statistics</i>	Shows IP protocol statistics
<i>ICMP Statistics</i>	Shows ICMP protocol statistics
<i>TCP Statistics</i>	Shows TCP protocol statistics
<i>UDP Statistics</i>	Shows UDP protocol statistics
<i>RAM Disk Usage</i>	Shows the (A)CS file system
<i>System Information</i>	Shows information about the kernel, time, CPU, and memory

Configuration using Telnet

The AlterPath Console Server box comes with an IP address preconfigured on its Ethernet interface (192.168.160.10). To access that box using telnet:

Step 1: Connect Hub to workstation and (A)CS.

Your workstation and your (A)CS must be on the same physical LAN. Connect one RJ-45 cable from the Ethernet port of the (A)CS to a spare port from a hub, and another

Chapter 2 - Installation and Configuration

cable from another spare port of that same hub to the workstation used to manage the servers.

Step 2: Add route pointing to the (A)CS IP.

From the workstation issue a command to add a route pointing to the network IP address of the (A)CS (192.168.160.0) accessed through the workstation's Ethernet interface.

For Linux, the command would be:

```
route add -net 192.168.160.0/24 gw <IP address assigned to the workstation's Ethernet interface>
```

Example: if the workstation has IP address 200.246.93.150 the command would be:

```
route add -net 192.168.160.0/24 gw 200.246.93.150
```

For Windows, the command would be:

```
route add 192.168.160.0 mask 255.255.255.0 <IP address assigned to the workstation's Ethernet interface>
```

Example: if the workstation has IP address 200.246.93.150 the command would be:

```
route add 192.168.160.0 mask 255.255.255.0 200.246.93.150
```

Step 3: Telnet to 192.168.160.10.

Step 4: Enter root as login name and tslinux as password.

Step 5: Type wiz and press Enter.

A wizard configuration screen will appear, asking you a series of questions.

Chapter 2 - Installation and Configuration

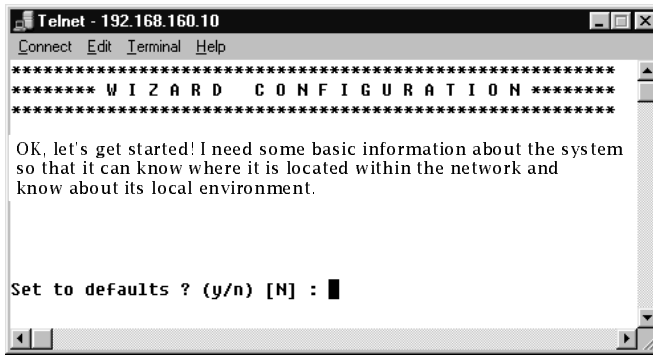


Figure 9: The initial wizard configuration screen

After you input the requested parameters you will receive a confirmation screen:

Your current configuration parameters are:

Hostname : CAS

System IP : 192.168.160.10

Domain name : cyclades.com

Primary DNS Server : 197.168.160.200

Gateway : 192.168.160.10

Network Mask : 255.255.255.0

If the parameters are correct, “Y” should be typed; otherwise, type “N” and then “C” when asked to change the parameters or quit the program. After the parameters are confirmed, the next question will be whether to save the configuration to flash. Select “Y” to make the new configuration permanent in non-volatile memory.

At this point you may lose your connection. Don't worry! The new configuration will be valid. The AlterPath Console Server is now configured as a CAS with its new IP address, with no authentication, and accepting telnet to the serial ports. You can telnet the CAS IP + serial port 1 with the following command:

```
telnet <IP you assigned> 7001
```

Chapter 2 - Installation and Configuration



Note. Serial port 1 is configured as 9600, 8N1 by default. The server connected to this serial port has to have the same configuration for its serial port.

After you confirm the basic parameters, you will be presented with the shell prompt. From there, either select to continue configuration using the vi editor or continue using a browser. For additional configuration, see [Chapter 3 - Additional Features](#) in this guide.

The Installation and Configuration Process

Task 1: Connect the AlterPath Console Server to the Network and other Devices

Power Users

Connect a PC or terminal to the AlterPath Console Server using the console cable. If using a PC, HyperTerminal can be used in the Windows operating system and Kermit or Minicom in the UNIX operating system. When the AlterPath Console Server boots properly, a login banner will appear. Log in as *root* (default password is *tslinux*). A new password should be created as soon as possible. The terminal parameters should be set as follows:

- Serial Speed: 9600 bps
- Data Length: 8 bits
- Parity: None
- Stop Bits: 1 stop bit
- Flow Control: none
- ANSI emulation

You may now skip to [Task 4: Edit the pslave.conf file](#).

Chapter 2 - Installation and Configuration



Important! Any configuration change must be saved in flash once validated. To save in **Flash** run `saveconf` (see [Task 7: Save the changes](#)). To validate/activate a configuration, run `signal_ras hup` (see [Task 5: Activate the changes](#)).



Note: If your terminal does not have ANSI emulation, select `vt100`; then, on the (A)CS, log in as root and switch to `vt100` by typing:

```
TERM=vt100;export TERM
```



Tip. We strongly recommend to use 9600 bps console speed. In case you need to use another speed please check [Appendix F - Software Upgrades and Troubleshooting](#).



Important! Always complete ALL the steps for your chosen configuration before testing or switching to another configuration.

New Users

If you are using a PC, you will be using HyperTerminal to perform the initial configuration of the AlterPath Console Server directly through your PC's COM port connected with the AlterPath Console Server console port. HyperTerminal, which comes with Windows 95, 98, Me, NT, 2K, and XP is often located under Start > Program > Accessories. HyperTerminal emulates a dumb terminal when your PC connects to the serial port (console port) of the AlterPath Console Server.

After the initial configuration through the HyperTerminal connection, you will be connecting your PC (or another terminal) to the AlterPath Console Server via an Ethernet connection in order to manage the (A)CS. The workstation used to access the (A)CS through telnet or ssh uses a LAN connection.

Chapter 2 - Installation and Configuration

These events can be summarized as follows:

- PC (Hyper terminal): COM port connects via serial cable to the (A)CS's console port.
- PC (Ethernet): Ethernet port connects via hub to the (A)CS's Ethernet port.
- Use the HyperTerminal to configure the box.
- Use the PC Ethernet to access the box as client (telnet/ssh).

Step 1: Plug the power cable into the AlterPath Console Server.

Insert the female end of the black power cable into the power socket on the AlterPath Console Server and the three-prong end into a wall outlet.



DANGER! To help prevent electric shock, plug the AlterPath Console Server into a properly grounded power source. The cable is equipped with a 3-prong plug to help ensure proper grounding. Do not use adapter plugs or remove the grounding prong from the cable. If you use an extension cable, use a 3-wire cable with properly grounded plugs.

Step 2: Connect the console cable.

You will be constructing a Console Cable out of the RJ-45 straight-through cable and the appropriate adapter provided in the product box. (There are four options: all adapters have an RJ-45 connector on one end, and either a DB25 or DB9 connector on the other end, male or female). Connect this cable to the port labeled "Console" on the AlterPath Console Server with the RJ-45 connector end, and connect the adapter end to your PC's available COM port. For more detailed information on cables, see [Appendix B - Cabling, Hardware, and Electrical Specifications](#).



Note: The modem cable is not necessary for a standard installation and configuration. Use it when the configuration is complete and you want to access the box remotely through a serial port.

Chapter 2 - Installation and Configuration

Step 3: Connect Hub to PC and the AlterPath Console Server.

Your workstation and (A)CS must be on the same physical LAN. Connect one RJ-45 cable from the Ethernet port of the (A)CS to the hub, and another from the hub to the workstation used to manage the servers.

Step 4: Install and launch HyperTerminal, Kermit or Minicom if not already installed.

You can obtain the latest update to HyperTerminal from:

<http://www.hilgraeve.com/http/download.html>

Task 2: Configure the COM Port Connection and Log In

Step 1: Select available COM port.

In HyperTerminal (Start > Program > Accessories), select File > Properties, and click the Connect To tab. Select the available COM port number from the Connection dropdown.

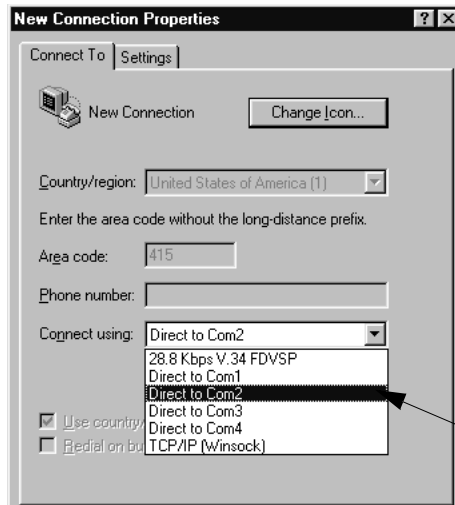


Figure 10: Choose a free COM port

Chapter 2 - Installation and Configuration

Step 2: Configure COM port.

Click the Configure button (hidden by the dropdown menu in the above figure). Your PC, considered here to be a “dumb terminal,” should be configured to use 9600 bps, 8 data bits, no parity, 1 stop bit, and no flow control (as shown in the following figure).

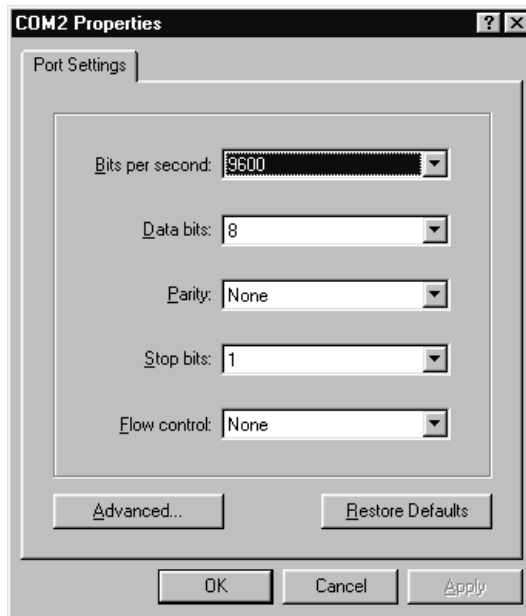


Figure 11: Port Settings

Step 3: Power on the AlterPath Console Server.

Step 4: Click OK on the Properties window.

You will see the Cyclades booting on your screen. After it finishes booting, you will see a login prompt.

Chapter 2 - Installation and Configuration

Task 3: Modify the System Files

When the AlterPath Console Server finishes booting, a prompt will appear (a flashing underline cursor) in your HyperTerminal window. You will modify the following Linux files to let the AlterPath Console Server know about its local environment:

```
/etc/hostname  
  
/etc/hosts  
  
/etc/resolv.conf  
  
/etc/network/st_routes
```

The four Linux files must be modified to identify the (A)CS and other devices it will be communicating with. The operating system provides the vi editor, which is described in [Appendix A - New User Background Information](#) for the uninitiated. The AlterPath Console Server runs Linux, a UNIX-like operating system, and those not familiar with it will want to refer to Appendix A.

Step 1: Type root and press Enter.

Step 2: At the password prompt, type *tslinux*.

Press Enter.

Step 3: Modify /etc/hostname.

In HyperTerminal, type “vi /etc/hostname” (without the quotes) and press Enter. Arrow over the existing text in the file, type “r” (for replace) and type the first number of the model of your AlterPath Console Server. (Or, you can replace the default naming convention with anything you’d like for your hostname.) When finished, press the Esc key, (to return to command mode), then type “:” (colon), and then “wq” and press Enter. This will save the file. (The only entry in this file should be the hostname of the AlterPath Console Server.) An example is shown in the following figure. (The HyperTerminal screen is shown in this first example for clarity, however, for the other Linux files we will modify, only the command line text will be shown.)

Chapter 2 - Installation and Configuration

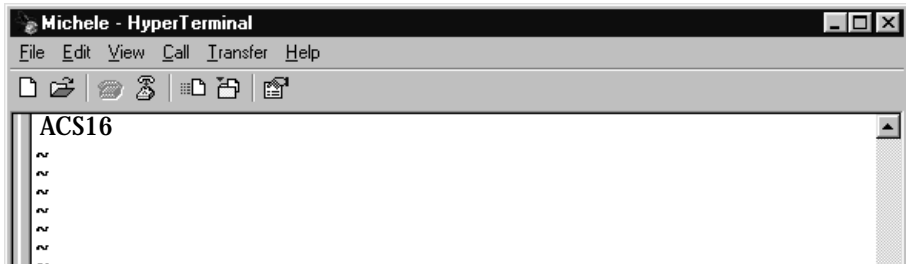


Figure 12: The `/etc/hostname` file with `hostname` typed in

Step 4: Modify `/etc/hosts`.

This file should contain the IP address for the Ethernet interface and the same hostname that you entered in the `/etc/hostname` file. It may also contain IP addresses and host names for other hosts in the network. Modify the file using the `vi` as you did in Step 1.

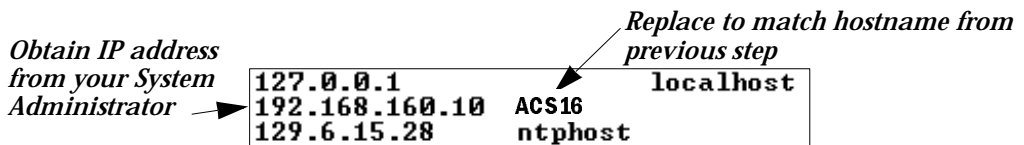


Figure 13: Contents of the `/etc/hosts` file

Step 5: Modify `/etc/resolv.conf`.

This file must contain the domain name and nameserver information for the network. Obtain the nameserver IP address from your Network Administrator. The default contents of this file are:

```
domain      mycompany.com
nameserver  200.200.200.2
```


Chapter 2 - Installation and Configuration

Step 6: Modify `/etc/network/st_routes`.

The fourth file defines static routes. In the console server example in [Figure 1: Console Access Server diagram](#) the router is a gateway router and thus its IP address is configured in this file to be the default gateway. Other static routes are also configured in this file. If you will be managing servers through a LAN, you don't need to alter this file. If you will be managing via Internet, you will be connecting through a router, and thus need to modify this file. You would get the IP address from your Network Administrator. The default contents of this file are:

```
route add default dev eth0
```

Step 7: Change password for root and new users.

The default `/etc/passwd` file has the user "root" with password "tlinux". You should change the password for user *root* as soon as possible. Before changing any password or adding new users you should also activate *shadow password*, if it is needed. The AlterPath Console Server has support for shadow password, but it is not active by default. To activate shadow password follow the steps listed below:

Step A: Create an empty file called `/etc/shadow`.

```
# cd /etc
# touch shadow
```

Step B: Add a temporary user to the system. It will be removed later.

```
# adduser boo
```

Step C: Edit the file *shadow*.

For each user in `passwd` file, create a copy of the line that begins with "boo:" in the `shadow` file, then replace "boo" with the user name. The line beginning with "root" must be the first line in the file `/etc/shadow`.

Step D: Edit the *passwd* file.

Replace the password in all password fields with an "x". The root's line will look like this:

```
"root:x:0:0:root:/root:/bin/sh"
  ^
  ^ password field
```

Chapter 2 - Installation and Configuration



Tip. Using the vi editor, put the cursor in the first byte after “root:”, then type “ct:x” plus <ESC>.

Step E: Remove the temporary user boo.

```
# deluser boo
```

Step F: Change the password for all users and add the new ones needed.

```
# passwd <username>
or
# adduser <username>
```

Step G: Edit `/etc/config_files` and add a line with `“/etc/shadow”`.

Task 4: Edit the `pslave.conf` file

This is the main configuration file (`/etc/portslave/pslave.conf`) that contains most product parameters and defines the functionality of the AlterPath Console Server. Only three parameters need to be modified or confirmed for a basic configuration:

- `conf.eth_ip`
- `all.authtype`
- `all.protocol`



Tip. You can do a find for each of these parameters in vi, once you open this file by typing `/ <your string>` to search the file downward for the string specified after the `/`.

A listing of the `pslave.conf` file with all possible parameters, as well as the files used to create other configurations from parameters in this file, is provided in [Appendix C - The `pslave Con`](#)

Chapter 2 - Installation and Configuration

[figuration File](#). Additional, optional modifications made to this file will depend on the configuration desired.

There are three basic types of parameters in this file:

- *conf.** parameters are global or apply to the Ethernet interface.
 - *all.** parameters are used to set default parameters for all ports.
 - *s#.** parameters change the default port parameters for individual ports.
- An *all.** parameter can be overridden by a *s#.** parameter appearing later in the *pslave.conf* file (or vice-versa).



Power Users: To find out what to input for these three parameters so that you can configure what you need, go the appropriate appendix, where you will find a complete table with an explanation for each parameter. You can use the templates from that same Appendix (*pslave.conf.cas*, etc.) as reference.

conf.eth_ip This is the IP address of the Ethernet interface. An example value would be:

200.200.200.1

all.authtype This parameter controls the authentication required by the AlterPath Console Server. The authentication required by the device to which the user is connecting is controlled separately. There are several authentication type options:

- *local* (authentication is performed using the */etc/passwd* file)
- *radius* (authentication is performed using a Radius authentication server)
- *TacacsPlus* (authentication is performed using a TacacsPlus authentication server)

Chapter 2 - Installation and Configuration

- *local/radius* (authentication is performed locally first, switching to Radius if unsuccessful)
- *none* (no authentication)
- *radius/local* (the opposite of the previous option)
- *RadiusDownLocal* (local authentication is tried only when the Radius server is down)
- *ldap* (authentication is performed against an ldap database using an ldap server. The IP address and other details of the ldap server are defined in the file `/etc/ldap.conf`)
- *kerberos* (authentication is performed using a kerberos server. The IP address and other details of the kerberos server are defined in the file `/etc/krb5.conf`)
- *local/TacacsPlus* (authentication is performed locally first, switching to TacacsPlus if unsuccessful)
- *TacacsPlus/local* (the opposite of the previous option)
- *TacacsPlusDownLocal* (local authentication is tried only when the TacacsPlus server is down)

An example value would be:

```
radius
```

all.protocol

For the console server configuration, the possible protocols are:

- *socket_server* (when telnet is used)
- *socket_ssh* (when ssh version one or two is used)
- *raw_data* (to exchange data in transparent mode – similar to *socket_server* mode, but without telnet negotiation, breaks to serial ports, etc.)

An example value would be:

```
socket_server
```

Chapter 2 - Installation and Configuration

The Authentication feature

See [Authentication](#) in [Chapter 3 - Additional Features](#).

Task 5: Activate the changes

Execute the following command in HyperTerminal to activate the changes:

```
signal_ras hup
```

Task 6: Test the configuration

Now you will want to make sure that the ports have been set up properly.

Step 1: Ping the (A)CS from a DOS prompt.

Open a DOS window, type in the following, and then press Enter:

```
ping <IP you assigned to the (A)CS>
```

An example would be:

```
ping 192.168.160.10
```

If you receive a reply, your (A)CS connection is OK. If there is no reply see [Appendix F - Software Upgrades and Troubleshooting](#).

Step 2: Telnet to the server connected to the first port of the AlterPath Console Server.

(This will only work if you selected `socket_server` as your `all_protocol` parameter.)

While still in the DOS window, type the following and then press Enter:

```
telnet <IP you assigned to the (A)CS> 7001
```

An example would be:

```
telnet 192.168.160.10 7001
```

If everything is configured correctly, a telnet session should open on the server connected to port 1. If not, check the configuration, follow the above steps again, and check [Appendix F - Software Upgrades and Troubleshooting](#).

Chapter 2 - Installation and Configuration

Task 7: Save the changes

Execute the following command in HyperTerminal to save the configuration:

```
saveconf
```

Task 8: Reboot the AlterPath Console Server

After rebooting, the initial configuration is complete.



Note: saveconf is equivalent to `tar -czf /proc/flash/script -T /etc/config_files` in standard Linux (saveconf must be used because tar on the (A)CS does not support the z flag).



Note: restoreconf does the opposite of saveconf, copying the contents of the `/proc/flash/script` file to the corresponding files in the ramdisk. The files on the ramdisk are overwritten. Restoreconf is run automatically each time the Alter-Path Console Server is booted.

Chapter 3 - Additional Features

Introduction

After the Configuration Wizard section in this chapter, each of the following sections is listed alphabetically and shows how to configure the option using vi, the custom Wizard (when available), browser, where appropriate, and the Command Line Interface (CLI), when available. This chapter contains the following sections:

- [Configuration Wizard - Basic Wizard](#)
- [Access Method](#)
- [Authentication](#)
- [Clustering](#)
- [CronD](#)
- [Data Buffering](#)
- [DHCP](#)
- [Dual Power Management](#)
- [Filters](#)
- [Generating Alarms](#)
- [Help](#)
- [NTP](#)
- [PCMCIA](#)
- [Ports Configured for Dial-in Access](#)
- [Ports Configured as Terminal Servers](#)
- [Serial Settings](#)
- [Session Sniffing](#)
- [SNMP](#)
- [Syslog](#)

Chapter 3 - Additional Features

- [Terminal Appearance](#)
- [Time Zone](#)



Note: If you add a user through the Web browser, the user does not actually get added to the list of users allowed to access the actual (A)CS unit.

Configuration Wizard - Basic Wizard

The configuration wizard application is a quicker and easier way to configure the AlterPath Console Server. It is recommended that you use this application if you are not familiar with the vi editor or if you just want to do a quick installation of the (A)CS.

The command *wiz* gets you started with some basic configuration. After executing this command, you can continue the configuration of the (A)CS using any browser or by editing system files with the vi editor. What follows are the basic parameters to get you quickly started. The files that will be eventually modified if you decide to save to flash at the end of this application are:

1. /etc/hostname
2. /etc/hosts
3. /etc/resolv.conf
4. /etc/network/st_routes
5. /etc/network/ifcfg_eth0
6. /etc/portslave/pslave.conf

Chapter 3 - Additional Features

Step 1: Enter the command *wiz*.

At the command prompt, type *wiz* in your (A)CS terminal to bring up the wizard. You will receive an initial prompt:

```
Set to defaults? (y/n) [N]:
```

Step 2: Press Enter or type *n* or *y*.

The default answer or value to any question is in the brackets. You can take one of three actions:

- Either just press the ENTER key to execute whatever is in between the brackets, or
- Type *n* to NOT reset the current configurations to the Cyclades defaults, or
- Type *y* to reset to Cyclades default configurations.

The next screen begins the configuration. There are instructions for using the wizard on each screen. There is also an explanation of each parameter before you are asked to configure it.



Tip. On most of the following configuration screens, the default or current value of the parameter is displayed inside brackets. Just press the ENTER key if you are satisfied with the value in the brackets. If not, enter the appropriate parameter and press ENTER.

If at any time, you want to exit the wizard or skip the rest of the configurations, press ESC. This will immediately display a summary of the current configurations for your verification before exiting the application. This will not work if you did not enter a valid choice for the parameter you are currently on.

Step 3: Enter Hostname and then press the Enter key.

This is an alias for your (A)CS that allows you to refer to the (A)CS by this name rather than its IP address. Enter hostname after the prompt:

```
Hostname[ CAS ]:
```

Chapter 3 - Additional Features

Step 4: Enter IP Address of your (A)CS and then press the Enter key.

This is the IP address of the (A)CS within your network. See your network administrator to obtain a valid IP address for the (A)CS.

```
IP of your system[192.168.160.10]:
```

Step 5: Enter Domain name and then press Enter.

Domain name locates or identifies your organization within the Internet.

```
Domain name[#]: cyclades.com
```

Step 6: Enter IP address of Domain Name Server and press Enter.

At the prompt, enter the IP address of the server that resolves domain names. Your domain name is alphabetical so that it is easier to remember. Every time you see the domain name, it is actually being translated into an IP address by the domain name server. See your network administrator to obtain this IP address for the domain name server.

```
Domain Name Server[#]: 192.168.160.200
```

Step 7: Enter Gateway IP address and press Enter.

The Gateway is a node on a network that serves as an entrance point into another network. See your network administrator to find out your organization's gateway address.

```
Gateway IP[eth0]: 192.168.160.10
```

Step 8: Enter Netmask and press Enter.

The Netmask is a string of 0s and 1s that mask or screen out the host part of an IP address so that only the network part of the address remains.

```
Netmask[255.255.255.0]:
```

Step 9: Review configuration parameters.

You will now have the parameters you just configured displayed back to you:

```
Your current configuration parameters are:
```

```
Hostname: CAS
```

```
System IP: 192.168.160.10
```

Chapter 3 - Additional Features

Domain Name: cyclades.com

Primary DNS Server: 192.168.160.200

Gateway: 192.168.160.10

Network Mask: 255.255.255.0

Are all these parameters correct (Y)es or (N)o [N]:

Step 10: Type *y*, or *n*, or press Enter.

Type *y* if all parameters are correct. Type *n* or just press ENTER if not all the parameters are correct and you want to go back and redo them.

If *n* is entered, this is displayed:

Type 'c' to go back and CORRECT the current configuration parameters or 'q' to QUIT:

Step 11: If you typed *n* in Step 10, type *c* or *q*.

As directed by the prompt, type *c* to go back to very beginning of this application to change the parameters. Type *q* to exit.

Step 12: If you typed *y* in Step 10, choose whether to save to flash.

Flash is a type of memory that will maintain the information saved on it even after the AlterPath Console Server is turned off. Once it is turned on again, the saved information can be recovered. If *y* is entered, the screen will display an explanation of what saving to flash means:

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time, thus making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the (A)CS even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the (A)CS.

Chapter 3 - Additional Features

Do you want to save your configurations to flash (Y/N) [N]:

Step 13: Type 'y' if you want to save to flash. Type 'n' if you don't want to save to flash.

You can now continue (A)CS configurations using the Web browser by typing in the IP address of the (A)CS.

Using the Wizard through your Browser

The Web interface supports wizards for serial ports configuration. The wizard is a useful tool that simplifies configuration of serial ports. The Web interface will access the following wizard files:

- /etc/portslave/pslave.wiz.cas (CAS)
- /etc/portslave/pslave.wiz.ts (TS)
- /etc/portslave/pslave.wiz.ras (Dial-in Access)

The step-by-step process to configuring ports for a specific profile appear in the following sections, and the exact screen flow begins with [Figure 14: Configuration and Administration page](#).

To summarize the process, the wizard configuration is started by first selecting the desired port(s) on the Port Selection page ([Figure 15: Port Selection page](#)), clicking Submit, and then selecting either the CAS, TS, or RAS profile buttons on the subsequent Serial Port Configuration Page ([Figure 16: Serial Port Configuration page - top](#)). Change the appropriate parameters, and then click the Submit button on the Serial Port Configuration Page. For most applications, the parameters to be changed are:

For CAS:

- Port Speed
- First RADIUS/TacacsPlus Authentication Server
- First Accounting Server
- RADIUS/TacacsPlus secret
- Protocol (if the protocol is Socket SSH, Socket Telnet, or Socket Raw)
- Socket Port (keep the "Incremented" option on)

Chapter 3 - Additional Features

For TS:

- Port Speed
- First RADIUS/TacacsPlus Authentication Server
- First Accounting Server
- RADIUS/TacacsPlus secret
- Protocol (if the protocol is Login, Rlogin, SSH, or Socket Client)
- Socket Port (write the TCP port for the protocol selected; keep the “incremented” option off)

For Dial-in access:

- First RADIUS/TacacsPlus Authentication Server
- First Accounting Server
- RADIUS/TacacsPlus secret
- Remote IP Address (keep the “Incremented” option on)

Access Method

Access method is how a user accesses a server connected with one of the serial ports on the AlterPath Console Server. You can access through telnet, SSH, or raw data. These methods are CAS-related. Access method also refers to users' access to the serial port, based on common users and administrative users. Accessing the AlterPath Console Server with a browser allows for both telnet and ssh methods. Additionally, you can view the server screen via the browser via a Java Applet. See the section in [Appendix F - Software Upgrades and Troubleshooting](#) called [“How to connect to serial ports from the browser” on page 285](#).

Chapter 3 - Additional Features

Parameters Involved and Passed Values

The parameters involved in configuring Access Method for CAS are as follows:

- all.ipno* This is the default IP address of the AlterPath Console Server 's serial ports. Any host can access a port using its IP address as long as a path to the address exists in the host's routing table. An example value would be 192.168.1.101+. The "+" indicates that the first port should be addressed as 192.168.1.101 and the following ports should have consecutive values.
- all.socket_port* In the CAS profile, this defines an alternative labeling system for the AlterPath Console Server ports. An example value would be 7001+. The "+" after the numerical value causes the serial interfaces to be numbered consecutively. In this example, serial interface 1 is assigned the port value 7001, serial interface 2 is assigned the port value 7002, etc. One example on how this could be used is in the case of all.protocol or s<n>.protocol socket_ssh and the port value (7001, 7002, etc), if supplied by the ssh client like username:port value, the ssh client will be directly connected with the serial interface.
- all.protocol* The possible protocols are telnet, ssh1/ssh2 or raw data:
socket_server = telnet protocol,
socket_ssh = ssh1/ssh2 protocol,
raw_data = used to exchange data in transparent mode. Raw_data is similar to socket_server mode but without telnet negotiation breaks to serial ports.
An example value would be socket_server.
- all.users* Restricts access to ports by user name (only the users listed can access the port or, using the character "!", all but the users listed can access the port.) In this example, the users joe, mark and members of user_group cannot access the port. A single comma and spaces/tabs may be used between names. A comma may not appear between the "!" and the first user name. The users may be local, Radius or TacacsPlus. User groups (defined with the parameter conf.group) can be used in combination with user names in the parameter list. Notice that these are common users, not administrators. Example value !joe, mark, user_group.

Chapter 3 - Additional Features

<i>all.poll_interval</i>	Valid only for protocols <code>socket_server</code> and <code>raw_data</code> . When not set to zero, this parameter sets the wait for a TCP connection keep-alive timer. If no traffic passes through the AlterPath Console Server for this period of time, the AlterPath Console Server will send a line status message to the remote device to see if the connection is still up. If not configured, 1000 ms is assumed (the unit for this parameter is ms). If set to zero, line status messages will not be sent to the socket client. Example value 0.
<i>all.tx_interval</i>	Valid for protocols <code>socket_server</code> and <code>raw_data</code> . Defines the delay (in milliseconds) before transmission to the Ethernet of data received through a serial port. If not configured, 100ms is assumed. If set to zero or a value above 1000, no buffering will take place. Example value 100.
<i>all.idletimeout</i>	Valid only for the CAS configuration (protocols <code>socket_server</code> , <code>socket_ssh</code> , and <code>raw_data</code>). Specifies how long (in minutes) a connection can remain inactive before it is cut off. If set to zero (the default), the connection will not time out. Example value 0.
<i>confgroup</i>	Used to group users to simplify configuration of the parameter <code>all.users</code> later on. This parameter can be used to define more than one group. Example value <code>group_name: user1, user2</code> .
<i>all.stty</i>	The <code>stty</code> command which can be issued to configure the serial port.
<i>s<n>.serverfarm</i>	Alias name given to the server connected to the serial port. <code>Server_connected</code> . Example value <code>Server_connected_serial1</code> .

Configuration for CAS

vi Method

The parameters described above must be changed by directly editing the `/etc/portslave/plsave.conf` file.

Chapter 3 - Additional Features

Browser Method

To configure Access Method with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

192.168.160.10

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

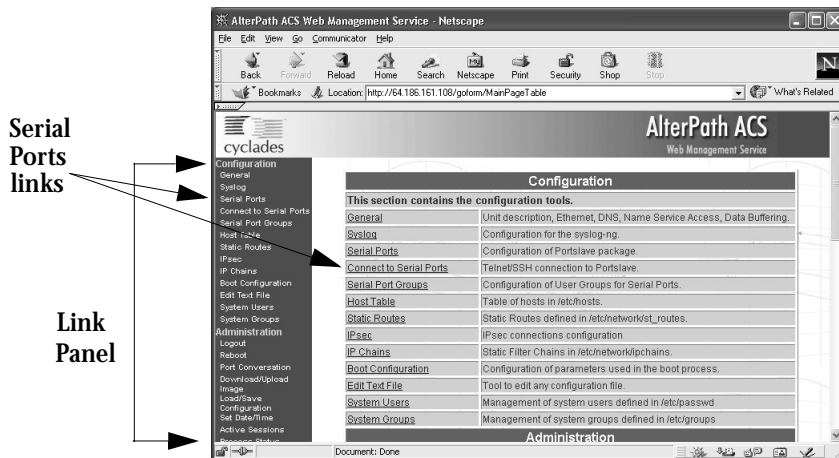


Figure 14: Configuration and Administration page

Step 3: Select the Serial Ports link.

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

Chapter 3 - Additional Features

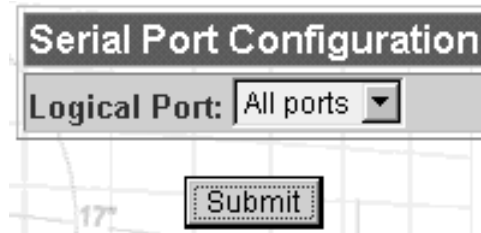


Figure 15: Port Selection page

Step 4: Select port(s).

On the Port Selection page, choose all ports or an individual port from the dropdown menu.

CAS profile button

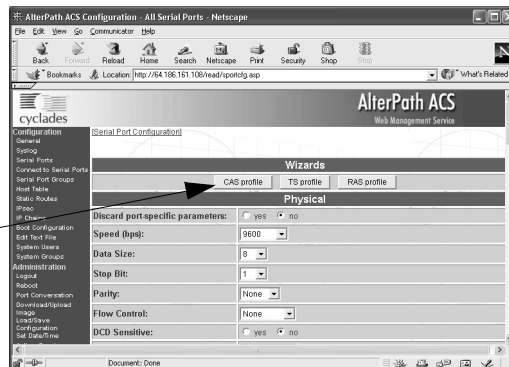


Figure 16: Serial Port Configuration page - top

Step 5: Click the CAS profile button.

Click the CAS profile button in the wizards section, then click the Submit button. This will take you to the Serial Port Configuration page.

Step 6: Scroll down to the Profile section.

You can change the settings for *all.ipno*, *all.socket_port*, and *all.protocol* in this section.

Chapter 3 - Additional Features

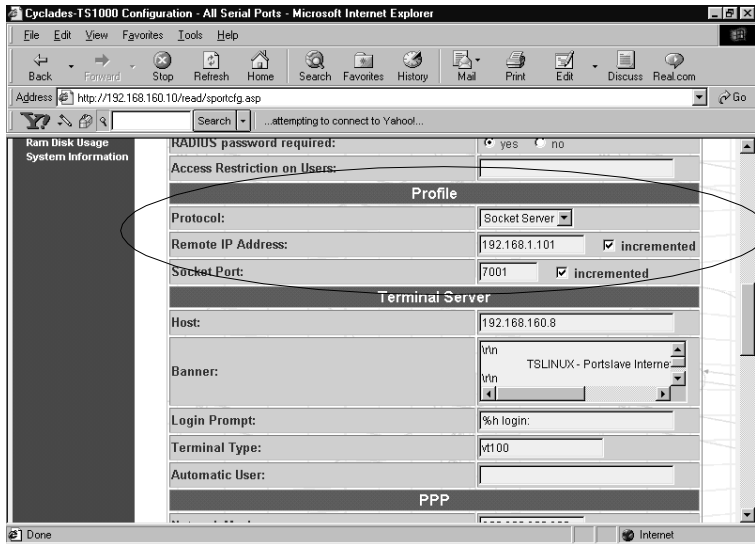


Figure 17: Profile Section of Serial Port Configuration page

Step 7: Scroll to the Authentication Section.

You can configure the parameter *all.users* here under Access Restriction on Users.

Step 8: Scroll to CAS Section.

You can configure the following parameters here:

- *all.poll_interval*
- *all.tx_interval*
- *all.idletimeout*

Step 9: Configure *s<n>.serverfarm*.

Scroll to the SSH section. As with the following two parameters, *s<n>.ipno* and *s<n>.socket_port*, there is one specific configuration per serial port. Each port can be named after the server or device connected to it. This makes the process of associating what is connecting to which port easier. This parameter will not appear on the configuration page when “All ports” is selected.

Chapter 3 - Additional Features

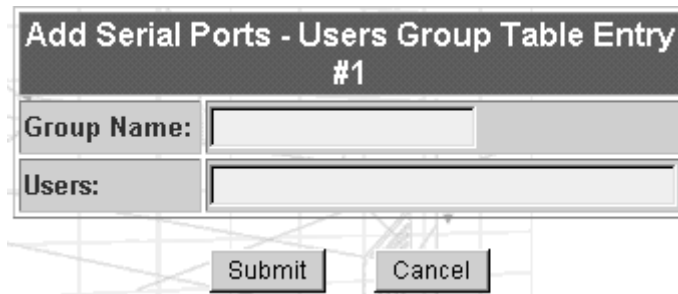
Step 10: Configure socket IP and socket port.

You can configure the socket profile of the following two parameters in the Profile section:

- `s<n>.ipno`
- `s<n>.socket_port`

Step 11: Click on the Serial Port Groups link on the Link Panel.

Click the Add Group button that appears. A Serial Ports - Users Group Table Entry page appears.



The image shows a web form titled "Add Serial Ports - Users Group Table Entry #1". The form contains two input fields: "Group Name:" and "Users:". Below the input fields are two buttons: "Submit" and "Cancel".

Figure 18: Serial Ports - Users Group Table Entry page

Step 12: Configure conf.group.

Fill in the Group Name and Users fields to configure the group.

Wizard Method

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Access Method custom wizard:

```
wiz --ac cas
```

This will bring up Screen 1:

Chapter 3 - Additional Features

Screen 1:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Set to defaults? (y/n) [N]:
```

Screen 2:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.IPNO - This is the default IP address of the system's serial ports. The '+' indicates that the first port should be addressed as 192.168.1.101 and the following ports should have consecutive values. Any host can access a port using its IP address as long as a path to the address exists in the host's routing table.

```
all.ipno[192.168.1.101+] :
```

ALL.SOCKET_PORT - This defines an alternative labeling system for the system ports. The '+' after the numerical value causes the interfaces (or ports) to be numbered consecutively.

Chapter 3 - Additional Features

(e.g. interface 1 of your system is assigned port 7001, interface 2 has the value 7002, etc.)

```
all.socket_port[7001+] :
```

Screen 3:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.PROTOCOL - The possible protocols are telnet, ssh1/ssh2, or raw data.

(e.g. socket_server -telnet protocol, socket_ssh -ssh1/ssh2 protocol, raw_data -used to exchange data in transparent mode; similar to socket_server mode but without telnet negotiation breaks to serial ports,.)

```
all.protocol[socket_server] :
```

ALL.USERS - Restricts access to ports by user name. Only the users listed can access the port, or using a '!', all but the users listed can access the port.

A single comma and spaces/tabs may be used between names. A comma may NOT appear between the '!' and the first user name. The users may be local, Radius or TacacsPlus. User groups (defined with the parameter conf.group) can be used in combination with user names in the parameter list. Notice that these are common users, not administrators.

Chapter 3 - Additional Features

(e.g. !joe, mark, grp1 -the users, Joe, Mark, and members of grp1, cannot access the port.)

```
all.users[#] :
```

Screen 4:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.POLL_INTERVAL - Valid for protocols socket_server and raw_data. When not set to 0, this parameter sets the wait for a TCP connection keep-alive timer. If no traffic passes through the system for this period of time, the system will send a line status message to the remote device to see if the connection is still up. If not configured, default is 1000ms. If set to 0, line status messages will not be sent to the socket client.

```
all.poll_interval[1000] :
```

ALL.TX_INTERVAL - Valid for protocols socket_server and raw_data. This parameter defines the delay (in milliseconds) before transmission to the Ethernet of data received through a serial port. If not configured, 100ms is assumed. If set to 0 or a value above 1000, no buffering will take place.

```
all.tx_interval[100] :
```

Chapter 3 - Additional Features

Screen 5:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.IDLETIMEOUT - This parameter specifies how long (in minutes) a connection can remain inactive before it is cut off. If set to 0 (the default), the connection will not time out.

all.idletimeout[0] :

CONF.GROUP - Used to combine users into a group. This simplifies the parameter, all.users. You can define more than one group. (e.g. groupName: user1, user2)

conf.group[#] :sales: john, jane

Would you like to create another group? (y/n) [N] :

Screen 6:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

Your current configuration parameters are:

(The ones with the '#' means it's not activated.)

```
all.ipno : 192.168.1.101+
all.socket_port : 7001+
```

Chapter 3 - Additional Features

```
all.protocol : socket_server
all.users : #
all.poll_interval : 1000
all.tx_interval : 100
all.idletimeout : 0
conf.group : #
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

If you type 'N':

Type 'c' to go back and CORRECT these parameters or 'q' to QUIT :

Typing 'c' repeats the application, typing 'q' exits the entire wiz application.

If you type 'Y':

Type 'c' to CONTINUE to set these parameters for specific ports or 'q' to QUIT :

Typing 'c' leads to Screen 7, typing 'q' leads to Screen 8.

Screen 7:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

You have 8 available ports on this system.

Type 'q' to quit, a valid port number[1-8], or anything else to refresh :



Note: The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. For “wiz -ac cas,” an additional parameter is asked: serverfarm. Typing 'q' leads to Screen 8.

Chapter 3 - Additional Features

Screen 8:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

CLI Method

To configure certain parameters for a specific serial port.

Step 1: Bring up the CLI.

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

Step 2: Type the following after the CLI prompt.

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

To configure the protocol. <string> is the type of protocol desired:

```
configure line <serial port number> protocol <string>
```

To configure the poll_interval:

Chapter 3 - Additional Features

```
configure line <serial port number> interval <number>
```

To configure the `socket_port`:

```
configure line <serial port number> socket <number>
```



Tip. You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> protocol  
<string> interval <number> socket <number>
```

Step 3: To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

Step 4: To activate your new configurations, type:

```
signal_ras hup
```

Configuration for TS

Parameters and Passed Values

For TS configuration, you will need to configure the following parameters:

all.host The IP address of the host to which the terminals will connect.
Example value 200.200.200.3.

all.protocol For the terminal server configuration, the possible protocols are
login (which requests username and password), rlogin (receives
username from the (A)CS and requests a password), telnet, ssh,
ssh2, or socket_client. See `all.socket_port` definition to see
when `all.protocol` should be configured as `socket_client`.
Example value rlogin.

Chapter 3 - Additional Features

all.socket_port

The `socket_port` is the TCP port number of the application that will accept connection requested by this serial port. That application usually is telnet (23). The `all.socket_port` (`s<n>.socket_port`) for the TS profile can be 23 (default value). This means that the TS will initiate a telnet session against a given host. If it is a different value, there will be pure, raw data between the client (TS for that serial port) and the host. The `all.protocol` (`s<n>.protocol` HAS to be configured as `socket_client`. In summary,
TS profile (`all.protocol` is `socket_client`)
raw mode (`all.socket_port` is NOT 23)

all.userauto (*unique to TS*)

Username used when connected to a UNIX server from the user's serial terminal.

all.issue

This text determines the format of the *login banner* that is issued when a connection is made to the AlterPath Console Server. `\n` represents a new line and `\r` represents a carriage return. Expansion characters can be used here.

Value for this Example:

```
\r\n\  
Welcome to terminal server %h port S%p \n\  
\r\n\  
\r\n\ Customer Support: 510-770-9727  
  
www.cyclades.com/\n\  
\r\n
```

all.prompt

This text defines the format of the *login prompt*. Expansion characters can be used here. Example value: `%h login.`

all.term

This parameter defines the *terminal type* assumed when performing rlogin or telnet to other hosts. Value for this example: `vt100`.

Chapter 3 - Additional Features

CLI Method

To configure certain parameters for a specific serial port.

Step 1: Bring up the CLI.

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

Step 2: Type the following after the CLI prompt.

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

To configure the protocol. <string> is the type of protocol desired:

```
configure line <serial port number> protocol <string>
```

To configure the socket_port:

```
configure line <serial port number> socket <number>
```



Tip. You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> protocol  
<string> socket <number>
```

Step 3: To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

Step 4: To activate your new configurations, type:

```
signal_ras hup
```

Chapter 3 - Additional Features

vi Method

The parameters described above must be changed by directly editing the `/etc/portslave/pslave.conf` file.

Browser Method

Step 1: Follow the steps 1 to 4 in the section titled Configuration for CAS, [“Browser Method” on page 64](#).

Step 2: Click the TS Profile button in the Wizard section.

Configure the following parameters:

Profile section: Protocol (telnet, ssh, rlogin or socket client)
 Socket port (23 for telnet, 22 for ssh, 513 for rlogin)

Terminal Server section: Host (the name or the IP address of the host)
 Automatic User

Step 3: Click the Submit button.

At this point, the configuration file is written in the RAMdisk.

Step 4: Make changes effective.

Go to the link Administration > Restart Processes and restart the `cy_ras` process.

Step 5: Save it in the flash.

Go to the link Administration->Load/Save Configuration and click the Save Configuration in flash button.

Wizard Method

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Access Method custom wizard:

```
wiz --ac ts
```

Screen 1 will appear.

Chapter 3 - Additional Features

Screen 1:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Set to defaults? (y/n) [N]:
```

Screen 2:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.PROTOCOL - Users can access the servers through the serial port using ssh, ssh2, telnet, login, rlogin, or socket_client. (e.g. login -requests username and password, rlogin -receives username from the system and requests a password, etc.)

all.protocol[rlogin] :

ALL.SOCKET_PORT - This defines the port(s) to be used by the protocols telnet and socket_client. For these two protocols a default value of 23 is used when no value is configured.

all.socket_port[23] :

Chapter 3 - Additional Features

Screen 3:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.USERAUTO - Username used when connected to a Unix server from the user's serial terminal.

all.userauto[#] :

Screen 4:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

Your current configuration parameters are:

(The ones with the '#' means it's not activated.)

```
all.protocol : rlogin
all.socket_port : 23
all.userauto : #
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

If you type 'N'

Type 'c' to go back and CORRECT these parameters or 'q' to QUIT :

Typing 'c' repeats the application, typing 'q' exits the entire wiz application

Chapter 3 - Additional Features

If you type 'Y'

Type 'c' to CONTINUE to set these parameters for specific ports or 'q' to QUIT :

Typing 'c' leads to Screen 5, typing 'q' leads to Screen 6.

Screen 5:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
You have 8 available ports on this system.
```

Type 'q' to quit, a valid port number[1-8], or anything else to refresh :



Note: The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 6.

Screen 6:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.
```

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

Chapter 3 - Additional Features

Configuration for Dial-in Access

Parameters and Passed Values

The parameters that need to be configured are:

<i>confpppd</i>	Location of the ppp daemon with Radius. Example value: /usr/local/sbin/pppd.
<i>conffacility</i>	This value (0-7) is the Local facility sent to the syslog. The file /etc/syslogng/syslog-ng.conf contains a mapping between the facility number and the action. Example value: 7.
<i>all.ipno</i>	This is the default IP address of the AlterPath Console Server 's serial ports. Any host can access a port using its IP address as long as a path to the address exists in the host's routing table. An example value would be 192.168.1.101+. The "+" indicates that the first port should be addressed as 192.168.1.101 and the following ports should have consecutive values.
<i>all.initchat</i>	Modem initialization string. Example value: TIMEOUT 10 "" \d\ \dATZ \OK\r\n-ATZ-OK\r\n "" "" ATMO OK\R\n "" \TIMEOUT 3600 RING "" \STATUS Incoming %p:I.HANDSHAKE "" ATA\TIMEOUT 60 CONNECT@ "" \ STATUS Connected %p:I.HANDSHAKE
<i>all.autoppp</i>	Options to auto-detect a ppp session. The cb-script parameter defines the file used for callback and enables negotiation with the callback server. Callback is available in combination with Radius Server authentication. When a registered user calls the (A)CS, it will disconnect the user, then call the user back. The following three parameters must be configured in the Radius Server: attribute Service_type(6): Callback Framed; attribute Framed_Protocol(7): PPP; attribute Callback_Number(19): the dial number (example: 50903300). Example value: %j novj \proxyarp modem asyncmap 000A0000 \noipx noccp login auth require-pap refusechap\mtu %t mru %t \cb-script /etc/portslave/ cb_script \plugin /usr/lib/libpsr.so

Chapter 3 - Additional Features

<i>all.pppopt</i>	PPP options when user has already been authenticated. Example value %i:%j novj \proxyarp modem asyncmap 000A0000 \noipx noccp mtu %t mru %t netmask%m \idle %I maxconnect %T \plugin /usr/lib/libpsr.so
<i>all.protocol</i>	For the Dial-in configuration, the available protocols are PPP, SLIP and CSLIP. Example value: ppp.
<i>s32.tty</i>	Example value: ttyS32.



Tip. Documentation about PPP options can be found on the Linux pppd man page.

vi Method

The parameters described above must be changed by directly editing the /etc/portslave/pslave.conf file.

Browser Method

For the serial ports you would have all the parameters described above but conf.*. To configure Access Method with your browser:

Step 1: Follow the steps 1 to 4 in the section titled Configuration for CAS, [“Browser Method” on page 64](#).

Step 2: Click the RAS Profile button in the Wizard section.

Step 3: Scroll down to the Profile section.

You can change the settings for *all.ipno* and *all.protocol* in this section.

Step 4: Scroll to the modem Section.

You can configure the parameter *all.initchat* here.

Step 5: Scroll to the PPP Section.

You can configure the parameter *all.autoppp* and *all.pppopt* here.

Chapter 3 - Additional Features

Step 6: Click on the Serial Port Groups link on the Link Panel.

Click the Add Group button that appears. A Serial Ports - Users Group Table Entry page appears.

Step 7: Configure socket **TTY**.

You can configure the socket profile of the s32.tty parameter in the Profile section.

CLI Method

To configure certain parameters for a specific serial port.

Step 1: Bring up the CLI.

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

Step 2: Type the following after the CLI prompt.

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

To configure the protocol. <string> is the type of protocol desired:

```
configure line <serial port number> protocol <string>
```



Tip. You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> protocol  
<string>
```

Step 3: To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

Step 4: To activate your new configurations, type:

```
signal_ras hup
```

Chapter 3 - Additional Features

Authentication

Authentication is the process of identifying an individual, usually based on a username and password. In security systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their identity. Authentication merely ensures that the individual is who he or she claims to be, but says nothing about the access rights of the individual. With the AlterPath Console Server, authentication can be performed locally, or with a remote Radius or Tacacs database.

Parameters Involved and Passed Values

The authentication feature utilizes the following parameters:

- all.authtype* Type of authentication used. There are several authentication type options:
- *local* (authentication is performed using the `/etc/passwd` file)
 - *radius* (authentication is performed using a Radius authentication server)
 - *TacacsPlus* (authentication is performed using a TacacsPlus authentication server)
 - *none*
 - *local/radius* (authentication is performed locally first, switching to Radius if unsuccessful)
 - *radius/local* (the opposite of the previous option),
 - *RadiusDownLocal* (local authentication is tried only when the Radius server is down)
 - *local/TacacsPlus* (authentication is performed locally first, switching to TacacsPlus if unsuccessful)

Chapter 3 - Additional Features

- *ldap* (authentication is performed against an ldap database using an ldap server. The IP address and other details of the ldap server are defined in the file `/etc/ldap.conf`)
- *kerberos* (authentication is performed using a kerberos server. The IP address and other details of the kerberos server are defined in the file `/etc/krb5.conf`)
- *TacacsPlus/local* (the opposite of the previous option), and
- *TacacsPlusDownLocal* (local authentication is tried only when the TacacsPlus server is down).

Note that this parameter controls the authentication required by the AlterPath Console Server . The authentication required by the device to which the user is connecting is controlled separately. Example value: radius.

all.authhost1 This address indicates the location of the Radius/TacacsPlus authentication server and is only necessary if this option is chosen in the previous parameter. A second Radius/TacacsPlus authentication server can be configured with the parameter *all.authhost2*. Example value 200.200.200.2.

all.accthost1 This address indicates the location of the Radius/TacacsPlus accounting server, which can be used to track how long users are connected after being authorized by the authentication server. Its use is optional. If this parameter is not used, accounting will not be performed. If the same server is used for authentication and accounting, both parameters must be filled with the same address. A second Radius/TacacsPlus accounting server can be configured with the parameter *all.accthost2*. Example value: 200.200.200.2.

all.authhost2 Example value: 200.200.200.3.

all.accthost2 Example value: 200.200.200.3.

all.radtimeout This is the timeout (in seconds) for a Radius/TacacsPlus authentication query to be answered. The first server (*authhost1*) is tried “radretries” times, and then the second (*authhost2*), if configured, is contacted “radretries” times. If the second also fails to respond, Radius/TacacsPlus authentication fails. Example value: 3.

Chapter 3 - Additional Features

- all.radretries* Defines the number of times each Radius/ TacacsPlus server is tried before another is contacted. The default, if not configured, is 5.
- all.secret* This is the shared secret (password) necessary for communication between the AlterPath Console Server and the Radius/TacacsPlus servers. Example value: rad-secret.

Configuration for CAS

vi Method

The parameters described above must be changed by directly editing the /etc/portslave/pslave.conf file.

Browser Method

To configure Authentication with your browser:

Step 1: Follow the steps 1 to 4 in the section titled Configuration for CAS, [“Browser Method” on page 64](#).

Step 2: Scroll to the Authentication section.

Scroll down to the Authentication section and configure the parameters in this section.

Wizard Method

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Authentication custom wizard:

```
wiz --auth
```

Screen 1 will appear.

Chapter 3 - Additional Features

Screen 1:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Set to defaults? (y/n) [N]:
```

Screen 2:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.AUTHTYPE - This parameter controls the authentication required by the system. Users' access to the server through the serial port is granted through the check of username and password locally or remotely. (e.g. none, local, TacacsPlus (note the capital 'T' in TacacsPlus), radius, ldap, kerberos, etc.

all.authtype[none] :

Chapter 3 - Additional Features



Note: If `authtype` is configured as “none,” “local,” “ldap,” or “kerberos” the application will skip immediately to the summary screen because the rest of the parameters pertain only if the system is configured to use a Radius or TacacsPlus server. Configurations for `ldap` and `kerberos` are done in `/etc/ldap.conf` and `/etc/krb5.conf`, respectively.

`ALL.AUTHHOST1` - This IP address indicates where the Radius or TacacsPlus authentication server is located.

```
all.authhost1[200.200.200.2] :
```

Screen 3:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

`ALL.ACCTHOST1` - This IP address indicates where the Radius or TacacsPlus accounting server is located. The accounting server can be used to track how long users are connected after being authorized by the authentication server.

```
all.accthost1[200.200.200.3] :
```


Chapter 3 - Additional Features

ALL.AUTHHOST2 - This IP address indicates where the SECOND Radius or TacacsPlus authentication server is located.

```
all.authhost2[200.200.200.2] :
```

Screen 4:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.ACCTHOST2 - This IP address indicates where the SECOND Radius or TacacsPlus accounting server is located.

```
all.accthost2[200.200.200.3] :
```

ALL.RADTIMEOUT- This is the timeout (in seconds) for a Radius or TacacsPlus authentication query to be answered.

```
all.radtimeout[3] :
```

Screen 5:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to

Chapter 3 - Additional Features

- deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.RADRETRIES - This defines the number of times each Radius or TacacsPlus server is tried before another is contacted.

all.radretries[5] :

ALL.SECRET - This is the shared secret necessary for communication between the system and the Radius or TacacsPlus servers.

all.secret[rad-secret] :

Screen 6:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****  
Your current configuration parameters are:  
(The ones with the '#' means it's not activated.)
```

```
all.authtype : none  
all.authhost1 : 200.200.200.2  
all.accthost1 : 200.200.200.3  
all.authhost2 : 200.200.200.2  
all.accthost2 : 200.200.200.3  
all.radtimeout : 3  
all.radretries : 5  
all.secret : rad-secret
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

If you type 'N'

Type 'c' to go back and CORRECT these parameters or 'q' to QUIT :

Chapter 3 - Additional Features

Typing 'c' repeats application, typing 'q' exits the entire wiz application

If you type 'Y'

Type 'c' to CONTINUE to set these parameters for specific ports or 'q' to QUIT :

Typing 'c' leads to Screen 7, typing 'q' leads to Screen 8.

Screen 7:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
You have 8 available ports on this system.
Type 'q' to quit, a valid port number[1-8], or anything
else to refresh :
```



Note: The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 8.

Screen 8:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N] :

Chapter 3 - Additional Features

CLI Method

To configure certain parameters for a specific serial port.

Step 1: Bring up the CLI.

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

Step 2: Type the following after the CLI prompt.

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

To configure authtype:

```
configure line <serial port number> authtype <string>
```



Tip. You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> authtype  
<string>
```

Step 3: To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

Step 4: To activate your new configurations, type:

```
signal_ras hup
```

Chapter 3 - Additional Features

Configuration for TS

The same `pslave.conf` parameters listed in the previous section are configured for a TS setup. You can use either the vi, Browser or CLI method if you want to configure parameters for a specific serial port.

Configuration for Dial-in Access

The same `pslave.conf` parameters listed in the previous section are configured for a Dial-in Access setup. You can use either the vi, Browser or CLI method if you want to configure parameters for a specific serial port.

Clustering

Clustering is available for the AlterPath Console Server with firmware versions 2.1.0 and up. It allows the stringing of Terminal Servers so that one master AlterPath Console Server can be used to access all AlterPath Console Servers on a LAN. The master AlterPath Console Server can manage up to 512 serial ports, so that the following can be clustered:

- 1 Master ACS16 + 31 slave ACS16s, or
- 1 Master ACS32 + 15 slave ACS32s

An example with one master ACS32 and two slave ACS16s is shown in the following figure.

Chapter 3 - Additional Features

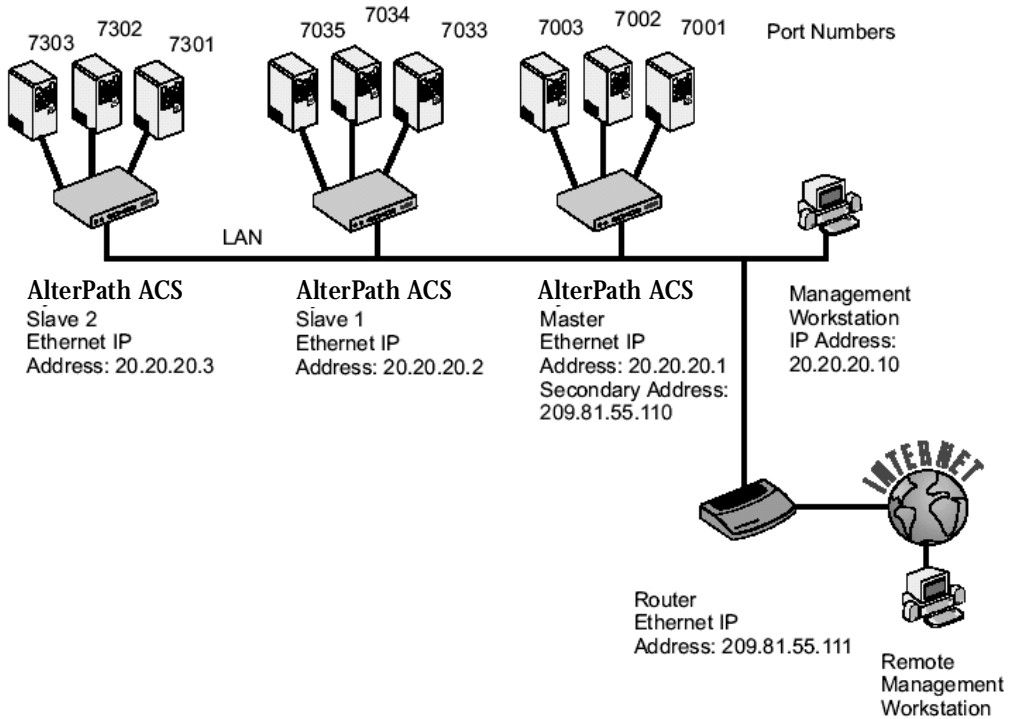


Figure 19: An example using the Clustering feature

Parameters Involved and Passed Values

The Master AlterPath Console Server must contain references to the Slave ports. The configuration described earlier for Console Access Servers should be followed with the following exceptions for the Master and Slaves:

Chapter 3 - Additional Features

Table 6: Master Cyclades Configuration (where it differs from the CAS standard)

Parameter	Description	Value for this example
conf.eth_ip	Ethernet Interface IP address.	20.20.20.1
conf.eth_ip_alias	Secondary IP address for the Ethernet Interface (needed for clustering feature).	209.81.55.110
conf.eth_mask_alias	Mask for secondary IP address above.	255.255.255.0
all.socket_port	This value applies to both the local ports and ports on slave AlterPath Console Server .	7001+
all.protocol	Depends on the application.	Socket_ssh or socket_server
all.authtype	Depends on the application.	Radius or local or none
s33.tty	This parameter must be created in the master (A)CS file for every slave port. Its format is: IP_of_Slave:[slave_socket_port] for non-master ports. In this case, the slave_socket_port value is not necessary because s33.socket_port is automatically set to 7033 by all.socket_port above.	20.20.20.2:7033
s33.serverfarm	An alias for this port.	Server_on_slave1_serial_s1
s33.ipno	This parameter must be created in the master (A)CS file for every slave port, unless configured using all.ipno.	0.0.0.0
s34.tty	See s33.tty.	20.20.20.2:7034
s34.serverfarm	An alias for this port.	Server_on_slave1_serial_s2

Chapter 3 - Additional Features

Table 6: Master Cyclades Configuration (where it differs from the CAS standard)

Parameter	Description	Value for this example
s34.ipno	See s33.ipno.	0.0.0.0
s35.tty	See s33.tty.	20.20.20.2:7035
s35.serverfarm	An alias for this port.	Server_on_slave1_serial_s3
s35.ipno	See s33.ipno.	0.0.0.0
etc. for s36-s64		
S65.tty	The format of this parameter is IP_of_Slave:[slave_socket_port] for non-master ports. The value 7301 was chosen arbitrarily for this example.	20.20.20.3:7301
S65.serverfarm	An alias for this port.	Server_on_slave2_serial_s1
S65.ipno	See s33.ipno.	0.0.0.0
S66.tty	See s65.tty	20.20.20.3:7302
S66.serverfarm	An alias for this port.	Server_on_slave2_serial_s2
S66.ipno	See s33.ipno.	0.0.0.0
S67.tty	See s65.tty.	20.20.20.3:7303
S67.serverfarm	An alias for this port.	Server_on_slave2_serial_s3
S67.ipno	See s33.ipno.	0.0.0.0
etc. for s68-s96		

Chapter 3 - Additional Features

The Slave AlterPath Console Server do not need to know they are being accessed through the Master AlterPath Console Server. (You are creating virtual terminals--virtual serial ports.) Their port numbers, however, must agree with those assigned by the Master.

Table 7: AlterPath Console Server configuration for Slave-1 (where it differs from the CAS standard)

Parameter	Value for this example
all.protocol	socket_server
all.authtype	none
conf.eth_ip	20.20.20.2
all.socket_port	7033+
all.authtype	none

Table 8: AlterPath Console Server configuration for Slave-2 (where it differs from the CAS standard)

Parameter	Value for this example
all.protocol	socket_server
all.authtype	none
conf.eth_ip	20.20.20.3
all.authtype	none
all.socket_port	7301+

To access ports from the remote management workstation, use telnet with the secondary IP address:

```
telnet 209.81.55.110 7001
```

to access the first port of the Master AlterPath Console Server.

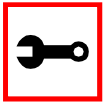
Chapter 3 - Additional Features

```
telnet 209.81.55.110 7033
```

to access the first port of Slave 1.

```
telnet 209.81.55.110 7065
```

to access the first port of Slave 2.



Note. Socket port 7065 is being used in the last example to access port 7301 in Slave 2.

Ssh can also be used from the remote management workstation:

```
ssh -l <username>:Server_on_slave2_serial_s3 209.81.55.110
```

to access the third port of Slave 2, or

```
ssh -l <username>:7069 209.81.55.110
```

to access the fifth port of Slave 2

Centralized Management - The Include File

The AlterPath Console Server allows centralized management through the use of a master `pslave.conf` file. Administrators should consider this approach to configure multiple AlterPath Console Server. Using this feature, each unit has a simplified `pslave.conf` file where a master include file is cited. This common configuration file contains information for all units, properly divided in separate sections, and would be stored on one central server. This file, in our example shown in [Figure 20: Example of Centralized Management](#), is `/etc/portslave/TScommon.conf`. It must be downloaded to each AlterPath Console Server.

Chapter 3 - Additional Features

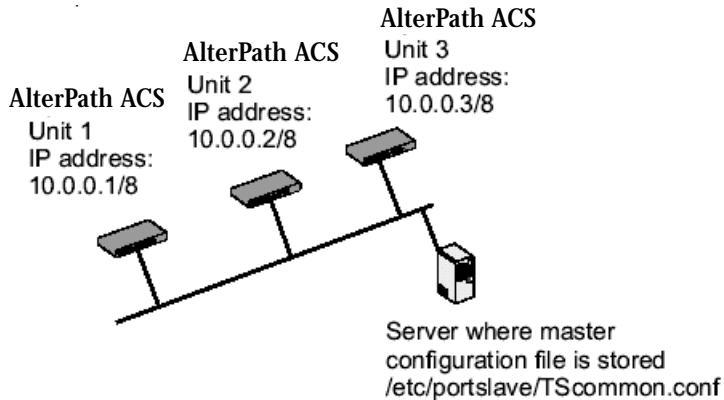


Figure 20: Example of Centralized Management

The abbreviated `pslave.conf` and `/etc/hostname` files in each unit, for the example are:

For the `/etc/hostname` file in *unit 1*:

```
unit1
```

For the `pslave.conf` file in *unit 1*:

```
conf.eth_ip 10.0.0.1
conf.eth_mask 255.0.0.0
conf.include /etc/portslave/TScommon.conf
```

For the `/etc/hostname` file in *unit 2*:

```
unit2
```

For the `pslave.conf` file in *unit 2*:

```
conf.eth_ip 10.0.0.2
conf.eth_mask 255.0.0.0
conf.include /etc/portslave/TScommon.conf
```

Chapter 3 - Additional Features

For the `/etc/hostname` file in *unit 3*:

```
unit3
```

For the `pslave.conf` file in *unit 3*:

```
conf.eth_ip 10.0.0.3
conf.eth_mask 255.0.0.0
conf.include /etc/portslave/TScommon.conf
```

The common include file for the example is:

```
conf.host_config unit1
<parameters for unit1 following the rules for pslave.conf>
conf.host_config unit2
<parameters for unit2 following the rules for pslave.conf>
conf.host_config unit3
<parameters for unit3 following the rules for pslave.conf>
conf.host_config.end
```

When this file is included, `unit1` would read only the information between “`conf.host_config unit1`” and `conf.host_config unit2`”. `Unit2` would use only the information between “`conf.host_config unit2`” and `conf.host_config unit3`” and `unit3` would use information after “`conf.host_config unit3`” and before `conf.host_config.end`.

Steps for using Centralized Configuration

Step 1: Create and save the `/etc/portslave/pslave.conf` and `/etc/hostname` files in each AlterPath Console Server.

Step 2: Execute the command `signal_ras hup` on each unit.

Chapter 3 - Additional Features

Step 3: Create, save, and download the common configuration.

Create and save the common configuration file on the server, then download it (probably using scp) to each unit. Make sure to put it in the directory set in the pslave.conf file (/etc/portslave in the example).

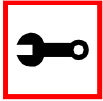
Step 4: Execute the command signal_ras hup on each unit again.

Step 5: Test each unit.

If everything works, add the line /etc/portslave/TScommon.conf to the /etc/config_files file.

Step 6: Save the file and close it.

Step 7: Execute the saveconf command.



Note: The included file /etc/portslave/TScommon.conf cannot contain another include file (i.e., the parameter conf.include must not be defined).

Also, <max ports of (A)CS> + N(+) is done same way as serial port.

Browser Method

To configure Clustering with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

192.168.160.10

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

Step 3: Select the Serial Ports link.

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

Chapter 3 - Additional Features

Step 4: Select port(s).

On the Port Selection page, choose all ports or an individual port to configure, from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

Step 5: Scroll down to the Profile section.

If you select the General page and go to the Ethernet section you'll find the common file field.

CronD

CronD is a service provided by the AlterPath Console Server system that allows automatic, periodically-run custom-made scripts. It replaces the need for the same commands to be run manually.

Parameters Involved and Passed Values

The following parameters are created in the `/etc/crontab_files` file:

- status
- user
- source

Configuration for CAS

vi Method

The files Crontab and Script shell are created and the file `/etc/crontab_files` is modified as indicated in the previous section.

Chapter 3 - Additional Features

To use cronD:

Step 1: Create the following two files for every process that it will execute:

<i>Crontab</i>	The file that specifies frequency of execution, the name of shell script, etc. It should be set using the traditional crontab file format.
<i>Script shell</i>	A script file with the Linux commands to be executed.

Step 2: Create a line in the file `/etc/crontab_files` for each process to be run.

Each line must contain the following three items:

<i>status (active or inactive)</i>	If this item is not active, the script will not be executed.
<i>user</i>	The process will be run with the privileges of this user, who must be a valid local user.
<i>source</i>	Pathname of the crontab file.

When the `/etc/crontab_files` file contains the following line:

```
active root /etc/tst_cron.src
```

and the `/etc/tst_cron.src` file contains the following line:

```
0-59 * * * * /etc/test_cron.sh
```

CronD will execute the script listed in `test_cron.sh` with root privileges each minute. Example files are in the `/etc` directory.

Step 3: Update the system.

The next step is to update the system with the modified data in the files above. Make sure the file named `/etc/config_files` contains the names of all files that should be saved to flash.

Chapter 3 - Additional Features

Step 4: Run `saveconf`.

The command `saveconf`, which reads the `/etc/config_files` file, should then be run. `saveconf` copies all the files listed in the file `/etc/config_files` from the ramdisk to `/proc/flash/script`.

Step 5: Reboot the AlterPath Console Server.

Browser Method

To configure CronD with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser, type:

```
192.168.160.10
```

Step 2: Log in.

Log in as root, pwd is `tslinux`. This will take you to the Configuration and Administration page.

Step 3: Click on the Edit Text File link.

Click on this link on the Link Panel or on the Configuration section of the Configuration and Administration page. (See [Figure 14: Configuration and Administration page](#). You can then pull up the appropriate file and edit it.

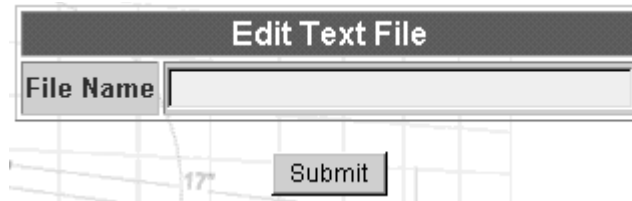


Figure 21: Edit Text File page

Configuration for TS

vi Method

This is done exactly as for CAS.

Chapter 3 - Additional Features

Configuration for Dial-in Access

vi Method

This is done exactly as for CAS.

Data Buffering

Introduction

Data buffering can be done in local files or in remote files through NFS. When using remote files, the limitation is imposed by the remote Server (disk/partition space) and the data is kept in linear (sequential) files in the remote Server. When using local files, the limitation is imposed by the size of the available ramdisk. You may wish to have data buffering done in file, syslog or both. For syslog, *all.syslog_buffering* and *conf.DB_facility* are the parameters to be dealt with, and *syslog-ng.conf* file should be set accordingly (please see [Syslog](#) for the *syslog-ng* configuration file). For the file *pslave.conf*, *all.data_buffering* is the parameter to be dealt with.

Conf.nfs_dat_buffering is the max file size per port. When commented, it indicates local data buffering. This parameter is a remote network file system where data buffering will be written, instead of using the default directory */var/run*. The directory tree to which the file will be written must be NFS-mounted. If data buffering is turned on for port 1, for example, the data will be stored in the file *ttyS1.data* (or *<serverfarm1>.data* if *s1.serverfarm* was configured) in the directory and server indicated by this variable. The remote host must have NFS installed and the administrator must create, export, and allow reading/writing to this directory. The size of this file is not limited by the value of the parameter *s1.data_buffering*, though the value cannot be zero since a zero value turns off data buffering.

Ramdisks

Since version 1.3.2 of the AlterPath Console Server software, additional ramdisks can be created and used, for example, to buffer data. This removed the previous 700 kbyte restriction for all (A)CS ports. Data buffering files are created in the directory */var/run/DB*. Previously, data buffering files were named *ttyS<nn>.data* (where *<nn>* is the port number). Now, if the parameter *s<nn>.serverfarm* is configured for the port *<nn>*, this name will be used. For example, if the serverfarm is called *bunny*, the data buffering file will be named *bunny.data*.

Chapter 3 - Additional Features

The shell script `/bin/build_DB_ramdisk` creates a 4 Mbyte ramdisk for the TS3000. Use the script `/bin/build_DB_ramdisk` as a model to create customized ramdisks for your environment. Any user-created scripts should be listed in the file `/etc/user_scripts` because `rc.sysinit` executes all shell scripts found there. This avoids changing `rc.sysinit` itself.

Linear vs. Circular Buffering

For local data buffering, this parameter allow users to buffer data in either a circular or linear fashion. Circular format (`cir`) is a revolving buffer file that is overwritten whenever the limit of the buffer size (set by `all.data_buffering`) is reached. In linear format (`lin`), data tranmission between the remote device and the serial port ceases once the 4k bytes Rx buffer in the kernel is reached. Then if a session is established to the serial port, the data in the buffer is shown (`dont_show_DBmenu` must be 2), cleared, and data transmission is resumed. Linear buffering is impossible if flow control is set to none. Default is `cir`.

Parameters Involved and Passed Values

Data Buffering uses the following parameters:

all.data_buffering

A non zero value activates data buffering (local or remote, according to what was configured in the parameter `conf.nfs_data_buffering`). If local data buffering, a file is created on the AlterPath Console Server; if remote, a file is created through NFS in a remote server. All data received from the port is captured in this file. If local data buffering, this parameter means the maximum file size (in bytes). If remote, this parameter is just a flag to activate (greater than zero) or deactivate data buffering. When local data buffering is used, each time the maximum is reached the oldest 10% of stored data is discarded, releasing space for new data (FIFO system) - circular file. When remote data buffering is used, there's no maximum file size other than the one imposed by the remote server - linear file. This file can be viewed using the normal UNIX tools (`cat`, `vi`, `more`, etc.). *Size is in bytes not kilobytes*. Example value 0.

Chapter 3 - Additional Features

conf.nfs_data_buffering

Remote Network File System where data captured from the serial port will be written instead of being written to the default directory “/var/run/DB.” The directory tree to which the file will be written must be NFS-mounted. If data buffering is turned on for port 1, for example, the data will be stored in the file `ttyS1.data` (or `<serverfarm1>.data` if `s1.serverfarm` was configured) in the directory indicated by this variable. The remote host must have NFS installed and the administrator must create, export and allow reading/writing to this directory. The size of this file is not limited by the value of the parameter `s1.data_buffering`, though the value cannot be zero since a zero value turns off data buffering. The size of the file is dependent on the NFS server only (hard drive, partition size, etc.).

all.DB_mode

Valid only when there is NO session (telnet/ssh/raw) established to the serial port; when a session is established to the serial port the data is always kept in a circular file. When configured as `cir` for circular format, the buffer is like a revolving file that is overwritten whenever the limit of the buffer size (as configured in `all.data_buffering` or `s<n>.data_buffering`) is reached. When configured as `lin` for linear format, once 4k bytes of the Rx buffer in the kernel is reached, a flow control stop (RTS off or XOFF—depending on how `all.flow` or `s<n>.flow` is set) is issued to prevent the serial port from receiving further data from the remote. Then when a session is established to the serial port, a flow control start (RTS on or XON) will be issued and data reception will then resume. If `all.flow` or `s<n>.flow` is set to `none`, neither linear nor circular buffering is possible. Default is `cir`.

all.syslog_buffering

When nonzero, the contents of the data buffer are sent to the syslog every time a quantity of data equal to this parameter is collected. The syslog level for data buffering is hard coded to level 5 (notice) and facility `conf.DB_facility`. The file `/etc/syslog-ng/syslog-ng.conf` should be set accordingly for the syslog-ng to take some action. Example value 0.

Chapter 3 - Additional Features

all.dont_show_DBmenu When zero, a menu with data buffering options is shown when a nonempty data buffering file is found. When 1, the data buffering menu is not shown. When 2, the data buffering menu is not shown but the data buffering file is shown if not empty. When 3, the data buffering menu is shown, but without the erase and show and erase options. Example value 0.

all.DB_timestamp Records the time stamp in the data buffering file (1) or not (0). If it is configured as 1, the software will accumulate input characters until it receives a CR and LF from the serial port or the accumulated data reaches 256 characters. Either way, the accumulated data will be recorded in the data buffering file along with the current time. The parameter *all.data_buffering* has to be with a non-zero value for this parameter to be meaningful. Ex. value 0.

Configuration for CAS

vi Method

Files created by the software:

- `ttyS<nn>.data`
- `s<nn>.serverfarm`

Files to be modified:

- `pslave.conf`
- `/etc/user_scripts`
- `syslog-ng.conf`

Browser Method

To configure Data Buffering with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

`192.168.160.10`

Chapter 3 - Additional Features

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

Step 3: Select the Serial Ports link.

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

Step 4: Select port(s).

On the Port Selection page, choose all ports or an individual port to configure, from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

Step 5: Scroll down to the Data Buffering section.

You can change the settings in this section.

Data Buffering	
Maximum Buffer Size (0-disabled):	<input type="text" value="0"/>
Data Buffering Mode:	CIR <input type="button" value="v"/>
Records the time stamp in the data buffering file:	<input type="radio"/> yes <input checked="" type="radio"/> no
Buffer size to send syslog (40 to 255, 0-disabled):	<input type="text" value="0"/>
Data Buffering Menu:	Show Menu <input type="button" value="v"/>
Alarm for Data Buffering:	Show Menu Don't Show Menu and Ignore DBfile Don't Show Menu and Show DBfile Show Short Menu
SSH	

Figure 22: Data Buffering section of the Serial Port Configuration page - menu dropdown

Chapter 3 - Additional Features

Data Buffering	
Maximum Buffer Size (0-disabled):	<input type="text" value="0"/>
Data Buffering Mode:	<input type="button" value="CIR"/> ▾
Records the time stamp in the data buffering file:	<input type="button" value="CIR"/> <input type="button" value="LIN"/> <input checked="" type="radio"/> no
Buffer size to send syslog (40 to 255, 0-disabled):	<input type="text" value="0"/>
Data Buffering Menu:	<input type="button" value="Show Menu"/> ▾
Alarm for Data Buffering:	<input type="radio"/> yes <input checked="" type="radio"/> no

Figure 23: Data Buffering section of the Serial Port Configuration page - mode dropdown

You can also configure Data Buffering on the General page (Link Panel > General link).

Data Buffering	
Remote NFS path:	<input type="text"/>
Data Buffering Facility:	<input type="button" value="local7"/> ▾

Figure 24: Data Buffering section of the General page

On this page you can choose whether NFS will be used or not.

Wizard Method

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Data Buffer custom wizard:

```
wiz --db
```

Chapter 3 - Additional Features

Screen 1:

```
*****
***** C O N F I G U R A T I O N W I Z A R D *****
*****
```

Data Buffering allows capturing of data received from the serial port and saving it into local files or remote files through NFS. Local file is circular and a maximum limit for its size (in bytes) is imposed by the available ramdisk. Remote file is sequential and its size is limited to the remote server's disk space.

Set to defaults ? (y/n) [N] :

Screen 2:

```
*****
***** C O N F I G U R A T I O N W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

CONF.NFS_DATA_BUFFERING - This parameter applies only if users choose to remotely buffer data. This is the remote directory name where data buffering will be written to instead of the default directory '/var/run'. If deactivated, data buffering will be done locally.

conf.nfs_data_buffering[#] :

ALL.DATA_BUFFERING - For local data buffering, this parameter represents the maximum file size in bytes allowed to be captured before it is discarded for new space. If re-

Chapter 3 - Additional Features

note this parameter is just a flag to either activate (any value greater than 0) or deactivate data buffering.

```
all.data_buffering[0] :
```

Screen 3:

```
*****  
***** C O N F I G U R A T I O N W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.DB_MODE - For local data buffering, this parameter allow users to buffer data in either a circular or linear fashion. Circular format (cir) is a revolving buffer file that is overwritten whenever the limit of the buffer size (set by all.data_buffering) is reached. In linear format (lin), data tranmission between the remote device and the serial port ceases once the 4k bytes Rx buffer in the kernel is reached. Then if a session is established to the serial port, the data in the buffer is shown (dont_show_DBmenu must be 2), cleared, and data transmission is resumed. Linear buffering is impossible if flow control is set to none. Default is cir.

```
all.DB_mode[cir] :
```

ALL.DONT_SHOW_DBMENU - When 0, a menu with data buffering options is shown when a non-empty data buffering file is found. When 1, the data buffering menu is not shown. When 2, the data buffering menu is

Chapter 3 - Additional Features

not shown but the data buffering file is shown if not empty. When 3, the data buffering menu is shown, but without the 'erase and show' and 'erase' options.

all.dont_show_DBmenu[0] :

Screen 4:

```
*****
***** C O N F I G U R A T I O N W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.DB_TIMESTAMP - Records the time stamp in the data buffering file (1) or not (0). In case it is configured as 1, the software will accumulate input characters until it receives a CR, an LF from the serial port, or the accumulated data reaches 256 characters. Either way, the accumulated data will be recorded in the data buffering file along with the current time. The parameter, all.data_buffering, has to be nonzero in order for this parameter to work.

all.DB_timestamp[0] :

ALL.SYSLOG_BUFFERING - This parameter is another option to data buffering. Users can also have syslog perform this function along with data buffering into files. When nonzero, the contents of the data buffer are sent to the syslog-ng every time a quantity of data equal to this parameter is collected. The syslog level for data buffering is hard coded to level 5 (notice) and facility

Chapter 3 - Additional Features

conf.DB_facility. The file /etc/syslog-ng/syslog-ng.conf should be set accordingly for the syslog-ng to take some action.

(Please see the 'Syslog-ng Configuration to use with Syslog Buffering Feature' section under Generating Alarms in Chapter 3 of the system's manual for the syslog-ng configuration file.)

```
all.syslog_buffering[0] :
```

Screen 5:

```
*****  
***** C O N F I G U R A T I O N W I Z A R D *****  
*****
```

Your current configuration parameters are:
(The ones with the '#' means it's not activated.)

```
conf.nfs_data_buffering : #  
all.data_buffering : 0  
all.DB_mode : cir  
all.dont_show_DBmenu : 0  
all.DB_timestamp : 0  
all.syslog_buffering : 0
```

Are these configuration(s) all correct (Y)es or (N)o [N] :



Note: The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 7.

Screen 6:

```
*****  
***** C O N F I G U R A T I O N W I Z A R D *****  
*****
```

You have 8 available ports on this system.

Chapter 3 - Additional Features

Type 'q' to quit, a valid port number[1-8], or anything else to refresh :

Screen 7:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]



Tip. If you did not type something greater than 0 for data_buffering, that means data_buffering isn't ON, so the wizard goes directly to Screen 3.



Tip. In all.dont_show_DBmenu, the difference between option 0 and option 2 is as follows: When 0, a menu with databuffering options is shown when a non-empty data buffering file is found. When 1, the data buffering menu is not shown. When 2, the data buffering menu is shown if not empty. When 3, the data buffering menu is shown, but without the “erase” and “show and erase” options.

Chapter 3 - Additional Features

DHCP

The DHCP (Dynamic Host Configuration Protocol) Client is available for firmware versions 1.2.x and above. DHCP is a protocol that allows network administrators to assign IP addresses automatically to network devices. Without DHCP (or a similar protocol like BOOTP), each device would have to be manually configured. DHCP automatically sends a new IP address to a connected device when it is moved to another location on the network. DHCP uses the concept of a fixed time period during which the assigned IP address is valid for the device it was assigned for. This “lease” time can vary for each device. A short lease time can be used when there are more devices than available IP numbers. For more information, see RFC 2131.

Parameter Involved and Passed Values

The DHCP client on the Ethernet Interface can be configured in two different ways, depending on the action the AlterPath Console Server should take in case the DHCP server does not answer the IP address request:

1. No action is taken and no IP address is assigned to the Ethernet Interface (most common configuration):
 - Set the global parameter `conf.dhcp_client` to 1.
 - Comment all other parameters related to the Ethernet Interface (`conf.eth_ip`, etc.).
 - Add the necessary options to the file `/etc/network/dhcpd_cmd` (some options are described below).
2. The AlterPath Console Server restores the last IP address previously provided in another boot and assigns this IP address to the Ethernet Interface:
 - Set the global parameter `conf.dhcp_client` to 2.
 - Comment all other parameters related to the Ethernet Interface (`conf.eth_ip`, etc.).
 - Add the following lines to the file `/etc/config_files`:

```
/etc/network/dhcpd_cmd
```

```
/etc/dhcpd-eth0.save
```

Chapter 3 - Additional Features

- Add the option “-x” to the factory default content of the file `/etc/network/dhcpd_cmd`:

```
/bin/dhcpd -x -c /bin/handle_dhcp
```

- Add all other necessary options to the file `/etc/network/dhcpd_cmd` (some options are described below). In both cases if the IP address of the AlterPath Console Server or the default gateway are changed, the AlterPath Console Server will adjust the routing table accordingly.

Two files are related to DHCP:

`/bin/handle_dhcp` The script which is run by the DHCP client each time an IP address negotiation takes place.

`/etc/network/dhcpd_cmd` Contains a command that activates the DHCP client (used by the `cy_ras` program). Its factory contents are:

```
/bin/dhcpd -c /bin/handle_dhcp
```

The options available that can be used on this command line are:

- D This option forces `dhcpd` to set the domain name of the host to the domain name parameter sent by the DHCP server. The default option is to NOT set the domain name of the host to the domain name parameter sent by the DHCP server.
- H This option forces `dhcpd` to set the host name of the host to the hostname parameter sent by the DHCP server. The default option is to NOT set the host name of the host to the hostname parameter sent by the DHCP server.
- R This option prevents `dhcpd` from replacing the existing `/etc/resolv.conf` file.



Note. Do not modify the `-c /bin/handle_dhcp` option.

Chapter 3 - Additional Features

Configuration for CAS

vi Method

Steps 1 and 2 under Parameters and Passed Values should be followed. You'll need to edit /etc/portslave/pslave.conf, comment some lines, etc.

Browser Method

To configure DHCP via your Web browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

192.168.160.10

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

Step 3: Click the General link on the Link Panel.

This takes you to the General page.

Step 4: Scroll down to the Ethernet port section.

You can activate DHCP Client in this section. Select the *active* radio button and click the Submit button at the bottom of the page.

Configuration for TS

vi Method

This is done exactly as for CAS.

Configuration for Dial-in Access

vi Method

This is done exactly as for CAS.

Chapter 3 - Additional Features

Dual Power Management

The AlterPath ACS comes with two power supplies which it can self-monitor. If either of them fails, two actions are performed: sounding a buzzer and generating a syslog message. This automanagement can be disabled (no actions are taken) or enabled (default), any time by issuing the commands:

```
signal_ras buzzer off
```

```
signal_ras buzzer on
```



Note: This section applies only to the AlterPath ACS. The AlterPath CS has a single power supply.

Parameters Involved and Passed Values

There are no parameters to be configured. However, if you want to generate alarms in case of a power failure, the `syslog-ng.conf` file must be changed.

Configuration for CAS

vi Method

Files to be changed:

```
/etc/syslog-ng/syslog-ng.conf
```

Browser Method

Follow the steps described in the section “Generating Alarms.”

Configuration for TS

vi Method

Same as for CAS.

Chapter 3 - Additional Features

Configuration for Dial-in Access

vi Method

Same as for CAS.

Filters

This feature is only available for firmware versions 2.1.0 and above. The AlterPath Console Server uses the Linux utility *iptables* to filter IP packets entering, leaving and passing through its interfaces. An ipchains tutorial is beyond the scope of this manual. For more information on ipchains, see the ipchains man page (not included with the AlterPath Console Server) or the how-to:

<http://www.netfilter.org>

or

<http://www.iptables.org>

Parameters Involved and Passed Values

The syntax of the *iptables* command is:

```
iptables - command chain [-s source] [-d destination]
[-p protocol] [-j target] [-i interface]
```

where *command* is one of the following:

- A** Add a condition or rule to the end of the chain. Note that the order in which a condition appears in a chain can modify its application and the first rule added to a chain is processed first, etc.
- D** Delete a condition from the chain. The condition must match exactly with the command's arguments to be deleted.
- R** Replace a condition in the chain.
- I** Insert a condition in a specified location in the chain.

Chapter 3 - Additional Features

- L* List all conditions in the chain.
- F* Flush (remove) all conditions in the chain.
- N* Create a new chain.
- X* Deletes a user-created chain.
- P* Policy applied for default handling

Chain is one of the following:

<i>INPUT</i>	Filters incoming packets.
<i>OUTPUT</i>	filters outgoing packets.
<i>FORWARD</i>	Filters packets which are not created by the AlterPath Console Server and are not destined to the AlterPath Console Server .
<i>user_created_chain</i>	A previously defined (or in the process of being defined) chain created using the N command is described above.

The output chain controls which packets are sent. A packet can be accepted by the input chain, but then rejected by the output chain. Likewise, the forward chain controls which packets will be routed. The input chain controls incoming packet filtering. The packet is either destined for the router or for another computer. In the latter case, the packet is processed by the forward chain. Packets that pass through the forward chain will then be processed by the output chain.

Source and *destination* have the following format:

```
[!]address[/ mask] [!][ port[: port]]
```

- !* Reverses the definition, resulting in the opposite.
- address* Host or network IP.
- port* Defines a specific port.
- port;port* Defines a range of ports.

Chapter 3 - Additional Features

If a source or destination is not specified then 0.0.0.0/0 is used.

Protocol is one of the following:

- tcp
- udp
- icmp
- all
- or a protocol number

(See the file /etc/protocols for a list.)

Target is one of the following:

- ACCEPT
- DROP
- The name of another chain

Interface is:

eth0 (The Ethernet interface is the only option on the AlterPath Console Server .) Lists do not need to be associated to an interface, so this option may be omitted.

To save changes made using the iptables command, execute *fwset*. This command will save the filter configuration in the file /etc/network/firewall and then save the file in flash memory.

To delete the changes made (before *fwset* is executed) execute *fwset restore* to return to the lists previously saved in /etc/network/firewall. Only the lists previously saved using *fwset* will then be defined. This command is executed at boot to invoke the last configuration saved. Another option is to edit the file /etc/network/firewall (or another file) directly, following the syntax defined in the file itself. If the file is edited in this way, the command *fwset* cannot be used to save and restore the configuration.

Chapter 3 - Additional Features

Use:

```
iptables-save > file_name
```

to save the lists in file_name,

```
updatefiles file_name
```

to save file_name to flash memory, and

```
iptables-restore < file_name
```

to restore the lists to the configuration in file_name.

An example of the use of the iptables filter for a Console Access Server

Referring to [Figure 1: Console Access Server diagram](#), if the administrator wishes to restrict access to the consoles connected to the AlterPath Console Server to a user on the workstation with IP address 200.200.200.4, a filter can be set up as shown below.

```
iptables -P INPUT ACCEPT
```

```
iptables -P OUTPUT ACCEPT
```

```
iptables -P FORWARD ACCEPT
```

```
iptables -A INPUT -p tcp -s ! 200.200.200.4 -d 0.0.0.0/0--destination-port 23 -j DROP
```

```
iptables -A INPUT -p tcp -s ! 200.200.200.4 -d 200.200.200.17001:7032 -j DROP
```

```
iptables -A INPUT -p tcp -s ! 200.200.200.4 -d 0.0.0.0/0--destination-port 22 -j DROP
```

Chapter 3 - Additional Features

Configuration for CAS

Browser Method

To configure filtering via your Web browser:

Step 1: Point your browser to 192.168.130.10.

Enter the (A)CS's IP address in your browser's address field.

Step 2: Log in.

Log in as *root*, with *tslinux* as a password. This will take you to the Configuration and Administration page. (See [“Configuration & Administration Menu page” on page 34](#))

Step 3: Click IPTables filter link.

On the Configuration section of this page, select the IPTables filter link. The following page will appear:

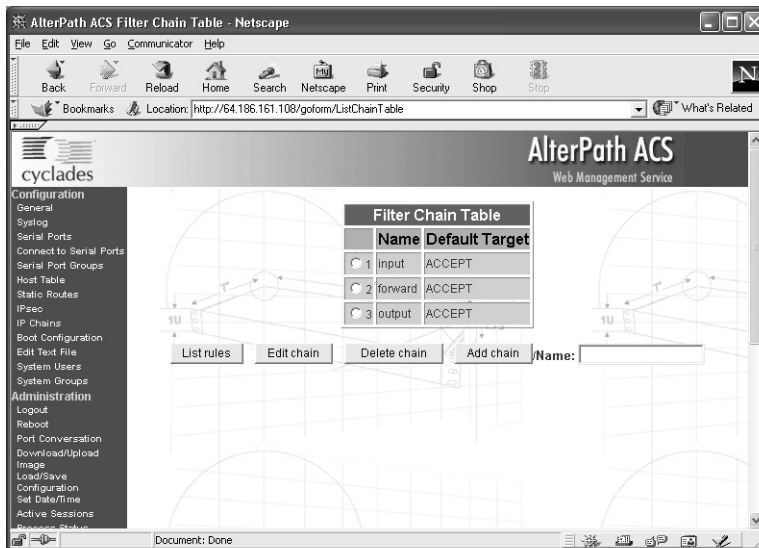


Figure 25: Page 1 of IPTables filtering

Chapter 3 - Additional Features

Step 4: Enter name of filter and select appropriate button.

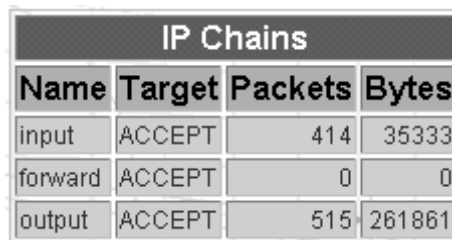
To create a filter, type in the name of the filter table in the Name box to the far right of the page, and then click the appropriate radio button to enter the default target. Then click the Add table button. Button functions are listed below:

<i>List rules</i>	When clicked, this button brings up a page of the selected IP chain filtering rules for the selected filter table.
<i>Edit table</i>	When clicked, this button brings up a page where you can edit the selected filter table.
<i>Delete table</i>	Lets you delete a selected filtering table.
<i>Add table</i>	Lets you add a filtering table.

A new filter will be added to the Port Table.

Step 5: Click the General Link.

If you click the General link on the Link Panel > IP Tables Filter > Information, you can view detailed information for each filter tabl.



IP Chains			
Name	Target	Packets	Bytes
input	ACCEPT	414	35333
forward	ACCEPT	0	0
output	ACCEPT	515	261861

Figure 26: IPTables filter Information page

General > Information also has IP Rules and IP Statistics links. IP Rules contains a table with rules on how to proceed when a datagram reaches the IP stack. IP Statistics has the following page:

Chapter 3 - Additional Features

IP Statistics		
Forwarding		1
DefaultTTL		64
InReceives		437
InHdrErrors		0
InAddrErrors		0
ForwDatagrams		0
InUnknownProtos		0
InDiscards		0
InDelivers		46
OutRequests		542
OutDiscards		0

Figure 27: IP Statistics page

Configuration for TS

vi Method

This is done the same as for CAS.

Configuration for Dial-in Access

vi Method

This is done the same as for CAS.

Generating Alarms

This feature helps the administrator to manage the servers. It filters the messages received by the serial port (the server's console) based on the contents of the messages. It then performs an action, such as sending an email or pager message. To configure this feature, you need to configure filters and actions in the syslog-ng.conf file. (You can read more about syslog-ng in the Syslog section.)

Chapter 3 - Additional Features

Port Slave Parameters Involved with Generating Alarms

- conf.DB_facility* This value (0-7) is the Local facility sent to the syslog-ng with data when syslog_buffering and/or alarm is active.
- all.alarm* When nonzero, all data received from the port is captured and sent to syslog-ng with INFO level and LOCAL[0+conf.DB_facility] facility.

vi Method

Files to be modified:

- pslave.conf
- syslog-ng.conf

Browser Method

To configure PortSlave parameters involved with syslog-ng and syslog-ng parameters with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

192.168.160.10

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

Step 3: Select the General link.

Click on the General link on the Link Panel to the left of the page in the Configuration section. This will take you to the General page.

Step 4: Scroll down to the Data Buffering section.

You can change the Data Buffering Facility value (conf.DB_facility). Click the Submit button.

Chapter 3 - Additional Features

Step 5: Select the Serial Ports link.

Click on the Serial Ports link on the Link Panel to the left of the page in the Configuration section. This will take you to the Port Selection page.

Step 6: Select port(s).

On the Port Selection page, choose all ports or an individual port to configure from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

Step 7: Scroll down to the Data Buffering section.

You can change the “Alarm for Data Buffering” (.alarm) value. Click the Submit button.

Step 8: Select the Syslog link.

Click on the Syslog link on the Link Panel to the left of the page in the Configuration section. This will take you to the Edit the Syslog-ng Configuration File page.

Wizard Method

The Alarm Generation custom wizard configures the ALL.ALARM parameter.

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Alarm Generation custom wizard:

```
wiz --al
```

Screen 1 (below) will appear.

Screen 1:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Set to defaults? (y/n) [N]:
```


Chapter 3 - Additional Features

Screen 2:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.ALARM - When non zero, all data received from the port are captured and sent to syslog-ng with DAEMON facility and ALERT level. The syslog-ng.conf file should be set accordingly, for the syslog-ng to take some action.

(Please see the 'Syslog-ng Configuration to use with Alarm Feature' section under Generating Alarms in Chapter 3 of the system's manual for the syslog-ng configuration file.)

all.alarm[0] :



Note: conf.DB_facility is configured under the syslog parameters (wiz - - sl).

Chapter 3 - Additional Features

Screen 3:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Your current configuration parameters are:
(The ones with the '#' means it's not activated.)

```
all.alarm : 0
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

If you type 'N'

Type 'c' to go back and CORRECT these parameters or 'q' to QUIT :

Typing 'c' repeats the application, typing 'q' exits the entire wiz application

If you type 'Y'

Type 'c' to CONTINUE to set these parameters for specific ports or 'q' to QUIT :

Typing 'c' leads to Screen 4, typing 'q' leads to Screen 5.

Screen 4:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

You have 8 available ports on this system.

Type 'q' to quit, a valid port number[1-8], or anything else to refresh :



Note: The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 5.

Chapter 3 - Additional Features

Screen 5:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

Syslog-ng Configuration to use with Alarm Feature

This configuration example is used for the alarm feature.

Step 1: Configure the pslave.conf file parameter.

In the pslave.conf file the parameters of the alarm feature are configured as:

```
all.alarm 1  
  
conf.DB_facility 2
```

Step 2: Add lines to syslog-ng.conf.

The syslog-ng.conf file needs these lines:

```
# local syslog clients  
  
source src { unix-stream("/dev/log"); };  
  
# To filter ALARM message with the string "kernel panic" :  
  
filter f_kpanic {facility(local2) and level(info) and  
match("ALARM") and match("kernel panic"); };  
  
# To filter ALARM message with the string "root login" :
```

Chapter 3 - Additional Features

```
filter f_root { facility(local2) and level(info) and
match("ALARM") and match("root login"); };

# To send e-mail to z@none.com (SMTP's IP address 10.0.0.2)
# from the e-mail address a@none.com with subject "ALARM".
# The message will carry the current date, the hostname
# of this unit and the message that was received from the
source.

destination d_maill {

    pipe("/dev/cyc_alarm"

        template("sendmail -t z@none.com -f a@none.com -s
\"ALARM\" -m \"$FULLDATE $HOST $MSG\" -h 10.0.0.2"));

};

# Example to send a pager to phone number 123 (Pager server
at 10.0.0.1) with message

# carrying the current date, the hostname of this (A)CS and
the message that was received from the source :

destination d_pager {

pipe("/dev/cyc_alarm"

template("sendsms -d 123 -m \"$FULLDATE $HOST $MSG\"
10.0.0.1"));

};

# Example to send a Link Down trap to server at 10.0.0.1 with
message carrying the current

# date, the hostname of this unit and the message that
received from the source :

destination d_trap {

pipe("/dev/cyc_alarm"
```

Chapter 3 - Additional Features

```
template("snmptrap -v 1 -c public 10.0.0.1 \"\" \"\" 2 0 \"\" \"\"
\
.1.3.6.1.2.1.2.2.1.2.1 s \"$FULLDATE $HOST $MSG\" "););
};

# To send e-mail and snmptrap if message received from local
syslog client has the string "kernel panic" :

log { source(sysl); filter(f_kpanic); destination(d_maill);
destination(d_trap); };

# To send e-mail and pager if message received from local
syslog client has the string

# "root login":

log { source(sysl); filter(f_root); destination(d_maill);
destination(d_pager); };
```

Alarm, Sendmail, Sendsms and Snmptrap

Alarm

This feature is available only for the Console Server Application. The (A)CS sends messages using pager, e-mail, or snmptrap if the serial port receives messages with specific string. To configure this feature:

Step 1: Activate alarm in Portslave configuration file.

Parameter `all.alarm` - 0 inactive or <> 0 active.

Step 2: Configure filters in the syslog-ng configuration file.

```
filter f_alarm { facility(local[0+conf.DB_facility]) and
level(info) and match("ALARM") and match("<your string>"); }
;
```

Example: to filter the ALARM message with the string "kernel panic" (conf.DB_facility is configured with value 1):

```
filter f_kpanic {facility(local1) and level(info) and
match("ALARM") and match ("kernel panic"); };
```

Chapter 3 - Additional Features

Example: to filter the ALARM message with the string “root login” :

```
filter f_root { facility(local1) and level(info) and
match("ALARM") and match("root login"); };
```

Step 3: Configure actions in syslog-ng configuration file.

(See more details in syslog-ng examples.)

Example: alarm is active and if the serial port receives the string “kernel panic,” one message will be sent to the pager.

```
log (source(sysl); filter(f_kpanic); destination(d_pager); );
```

To send e-mail:

```
destination d_mail { pipe("/dev/cyc_alarm" template("send-
mail <pars>"));};
```

To send a pager message:

```
destination d_pager {pipe("/dev/cyc_alarm" template("sendsms
<pars>"));};
```

To send snmptrap:

```
destination d_trap {pipe("/dev/cyc_alarm" template("snmptrap
<pars>"));};
```

Step 4: Connect filters and actions in the syslog-ng configuration file.

Example: alarm is active and if the serial port receives the string “kernel panic”, one message will be sent to the pager.

```
log (source(sysl); filter(f_kpanic); destination(d_trap);
destination(d_pager); );
```

Sendmail

Sendmail sends a message to a SMTP server. It is not intended as a user interface routine; it is used only to send pre-formatted messages. Sendmail reads all parameters in the command

Chapter 3 - Additional Features

line. If the SMTP server does not answer the SMTP protocol requests sent by sendmail, the message is dropped.

Synopsis:

```
sendmail -t <name>[,<name>] [-c <name> [,<name>]] [-b <name>
[,<name>]] [-r <name>] -f <name> -s <text> -m <text> -h <SMTP
server> [-p <smtp-port>]
```

where:

<i>-t <name>[,<name>]</i>	“To:” Required. Multi-part allowed (multiple names are separated by commas). Names are expanded as explained below.
<i>[-c <name> [,<name>]]</i>	“Cc:” Optional. Multi-part allowed (multiple names are separated by commas).
<i>[-b <name> [,<name>]]</i>	“Bcc:” Optional. Multi-part allowed (multiple names are separated by commas).
<i>[-r <name>]</i>	“Reply-To:” Optional. Use the Reply-To: field to make sure the destination user can send a reply to a regular mailbox.
<i>-f <name></i>	“From:” Required.
<i>-s <text></i>	“Subject:” Required.
<i>-m <text></i>	“body” The message body.
<i>-h <SMTP server></i>	Required. IP address or name of the SMTP server.
<i>[-p <SMTP port></i>	Optional. The port number used in the connection with the server. Default: 25.
<i><name></i>	Any email address.
<i><text></i>	A text field. As this kind of field can contain blank spaces, please use the quotation marks to enclose the text.

For example, to send e-mail to z@none.com (SMTP's IP address 10.0.0.2) from the e-mail address a@none.com with subject “sendmail test.”

Chapter 3 - Additional Features

```
sendmail -t z@none.com -f a@none.com -s "sendmail test" -m "Send-  
mail test. \n Is it OK???" -h 10.0.0.2
```

Sendsms

The `sendsms` is the Linux command line client for the SMSLink project. It accepts command line parameters that define the message to be sent, and transmits them to the SMS server process running on the designated server. The `sendsms` was developed specifically for easy calling from shell scripts or similar situations.

Synopsis:

```
sendsms [-r] [-g] [-v] -d dest (-m message or -f msgfile)  
[-u user] [-p port] server
```

where:

- r** Reporting. Additional info will be included in the message printed on stderr (namely, the device name used by the server to send the SMS out, and the message ID attributed to the SMS by the module's SIM card). If any of these items is missing or can't be parsed, a value of "???" will be returned.
- g** Turns debugging on. Will output the entire dialog with the server on stderr (and more).
- h** Displays a short help message and exits.
- v** Displays version information and exits.

Chapter 3 - Additional Features

-d dest

Required. The GSM network address (i.e. phone number) of the mobile phone the message is to be sent to. Supported format is: [int. prefix - country code] area code - phone number. The international prefix can be either “+” or “00” (or any other value supported by the GSM network provider the server is subscribed to). Some separation characters can be used to beautify the number, but they are purely cosmetic and will be stripped by the server. Those characters are [./-]. The pause character (',') is not supported. Regarding the international country code, don't forget that its necessity is to be considered respective to the SMS gateway location (the host this client program is connecting to), not the location where the client is run from. In case of doubt, please contact the SMS server administrator for your network. Please always include the area code (even when sending to a destination in the same “area”, i.e., on the same network). The number without the area code, though syntactically correct and accepted by the network, may never get delivered.

-m message

Required (Use one and only one of “-m” or “-f”). The text of the message to be sent. Unless made up of a single word, it will have to be quoted for obvious reasons. Maximum length is 160 characters. A longer message will be truncated (you will be warned about it), but the message will still be sent. At the present time, only 7bit ASCII is supported for the message text.

-f msgfile

Required (use one and only one of “-m” or “-f”). The name of a text file where the message to send is to be read from. This file can contain multiple lines of text (they will be concatenated), but its total length can't exceed 160 characters. A longer text will be truncated (you will be warned about it), but the message will still be sent. The special file '-' means that input will be read from stdin. At the present time, only 7-bit ASCII is supported for the message text.

-u user

Optional. The server module requires the user to identify her/himself for logging purposes. No authentication is performed on this information, however. If this parameter is omitted, sendsms will send the UNIX username of the current user. This parameter allows you to override this default behavior (might be useful in the case of automated sending).

Chapter 3 - Additional Features

-p port Optional. Communication port on the target server. If provided here, this value will be used to connect to the server. If omitted, the client will query the local system for the port number associated with the “well known service” sms (as defined in */etc/services*). If that doesn't return an answer, the compiled-in default value 6701 will be used.

server Required. The host name or IP address of the computer where the SMS gateway server process is running. By default, this server will be listening on TCP port 6701.

Upon success (when the server module reports that the message was successfully sent), *sendsms* returns 0. When a problem occurs, a non zero value is returned. Different return values indicate different problems. A return value of 1 indicates a general failure of the client program.

COPYRIGHT: SMSLink is (c) Les Ateliers du Heron, 1998 by Philippe Andersson.

Example to send a pager message to phone number 123 (Pager server at 10.0.0.1) with message:

```
sendsms -d 123 -m "Hi. This is a test message send from (A)CS using  
sendsms" 10.0.0.1
```

Snmpttrap

Snmpttrap is an SNMP application that uses the TRAP-PDU Request to send information to a network manager. One or more fully qualified object identifiers can be given as arguments on the command line. A type and a value must accompany each object identifier. Each variable name is given in the format specified. If any of the required version 1 parameters—enterprise-oid, agent and uptime—are specified as empty, it defaults to “.1.3.6.1.4.1.3.1.1”, host-name, and host-uptime respectively.

Synopsis

```
snmptrap -v 1 [-Ci] [common arguments] enterprise-oid agent  
generic-trap specific-trap uptime [objectID type value]...
```

```
snmptrap -v [2c|3] [-Ci] [common arguments] uptime trap-oid  
[objectID type value]...
```

Chapter 3 - Additional Features

where:

<i>-Ci</i>	Optional. It sends INFORM-PDU.
<i>common arguments</i>	Required. They are: SNMP server IP address and community.
<i>enterprise-oid</i>	Required, but it can be empty (").
<i>agent</i>	Required, but it can be empty ("). The agent name.
<i>generic-trap</i>	The generic trap number: 2 (link down), 3 (link up), 4 (authentication failure), ...
<i>specific-trap</i>	Required. The specific trap number.
<i>uptime</i>	Required.
<i>[objectID type value]</i>	Optional. objectID is the object oid. You want to inform its value to server.

If the network entity has an error processing the request packet, an error packet will be returned and a message will be shown, helping to pinpoint in what way the request was malformed. If there were other variables in the request, the request will be resent without the bad variable.

For example, to send a Link Down trap to server at 10.0.0.1 with interfaces.iftable.ifentry.ifde-scr:

```
snmptrap -v 1 -c public 10.0.0.1 " 2 0 " .1.3.6.1.2.1.2.2.1.2.1  
s "(A)CS: serial port number 1 is down"
```

<i>-Ci</i>	Optional. It sends INFORM-PDU.
<i>common arguments</i>	Required. They are: SNMP server IP address and community.
<i>enterprise-oid</i>	Required, but it can be empty (").

Chapter 3 - Additional Features

Help

Help Wizard Information

Synopsis: `wiz [--OPTIONS] [--port <port number>]`



Note: Make sure there are two hyphens before any of the options listed on the following table.

Table 9: General Options for the Help Wizard

Option	Description
<i>auth</i>	Configuration of authentication parameters
<i>tl</i>	Configuration of terminal login display parameters
<i>al</i>	Configuration of alarm parameter
<i>db</i>	Configuration of data buffering parameters
<i>snf</i>	Configuration of sniffing parameters
<i>sl</i>	Configuration of syslog parameters
<i>tso</i>	Configuration of other parameters specific to the TS profile
<i>ac</i> <cas or ts>	Configuration of access method parameters
<i>sset</i> <cas or ts>	Configuration of serial setting parameters
<i>all</i> <cas or ts>	Configuration of all parameters
<i>help</i>	Print this help message

Chapter 3 - Additional Features



Note: To directly configure a feature for a specific serial port, use the “-port <port number>” option after “wiz --[option].”

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Help custom wizard (you can also type `wiz -h`):

```
wiz --help
```

Help Command Line Interface Information

Synopsis 1

```
config configure line [serial port number] [options]
```

or

```
configure line [serial port number] [options]
```

(The comand above is valid only after entering into CLI mode. This is done by first just typing `config` at the terminal prompt. Then you will get a CLI prompt such as `config@hostname>>`. Once in the CLI mode, you eliminate the need to type `config` in all your CLI commands.)

Table 10: Help CLI Options - Synopsis 1

Option	Description
<i>ttys</i> <string>	Activate the serial port.
<i>protocol</i> <string>	Configuration of protocol for the serial port.
<i>interval</i> <number>	Configuration of poll_interval for the serial port.
<i>authtype</i> <string>	Configuration of authentication type for the serial port.

Chapter 3 - Additional Features

Table 10: Help CLI Options - Synopsis 1

Option	Description
<i>speed</i> <number>	Configuration of speed for the serial port.
<i>datsize</i> <number>	Configuration of datasize for the serial port.
<i>stopbits</i> <number>	Configuration of the stopbits for the serial port.
<i>parity</i> <string>	Configuration of the parity for the serial port.
<i>socket</i> <number>	Configuration of socket_port for the serial port.
<i>break</i> <string>	Configuration of break_sequence for the serial port.

There are also other options that configures network related parameters.

Synopsis 2

```
config configure ether [options]
```

or

```
configure ether [options]
```

(This synopsis is valid only after entering into CLI mode. This is done by first just typing `config` at the terminal prompt. Then, you will get a CLI prompt such as `config@hostname>>`.)

Table 11: Help CLI Options - Synopsis 2

Option	Description
<i>ip</i> <string>	Configuration of the IP of the Ethernet interface.
<i>mask</i> <string>	Configuration of the mask for the Ethernet network
<i>mtu</i> <number>	Configuration of the Maximum Transmission Unit size

Chapter 3 - Additional Features

Requesting Help for the CLI

There are two methods for requesting help for the CLI:

- To obtain general help on the format of CLI, type *config help* at the command prompt, or if you are already in the CLI, just type *help* after the CLI prompt.
- Help may be requested at any point in a command by entering a “?”. If nothing matches, the help list will be empty and you must backup until entering a “?” shows the available options.

For example:

- To find out possible commands that can come after *config*, type:

```
config ?
```

- To find out what parameters are configurable through CLI, type:

```
config configure line <serial port number> ?
```

NTP

The *ntpc* client is a *Network Time Protocol* (RFC-1305) client for UNIX- and Linux-based computers. In order for the AlterPath Console Server to work as a NTP client, the IP address of the NTP server must be set in the file `/etc/ntpc.conf`.

The script shell `/bin/ntpc.sh` reads the configuration file (`/etc/ntpc.conf`) and build the line command to call `/bin/ntpc` program.

Parameters Involved and Passed Values

The file `/etc/ntpc.conf` has the value of two parameters:

<i>NTPSERVER</i>	The IP address of the NTP server.
<i>INTERVAL</i>	Check time every interval seconds (default 300).

The data and time will be update from the NPT server according to the parameter options.

Chapter 3 - Additional Features

The ntpclient program has this syntax:

```
ntpclient [options]
```

Options:

<i>-c count</i>	Stop after count time measurements (default 0 means go forever).
<i>-d</i>	print diagnostics
<i>-h hostname</i>	NTP server host(mandatory).
<i>-i interval</i>	Check time every interval seconds.
<i>-l</i>	Attempt to lock local clock to server using adjtimex(2).
<i>-p port</i>	Local NTP client UDP port.
<i>-r</i>	Replay analysis code based on stdin.
<i>-s</i>	Clock set (if count is not defined this sets count to 1).

Configuration for CAS

vi Method

Files to be changed:

```
/etc/ntpclient.conf
```

Browser Method

To configure NTP with your browser:

Step 1: Point your browser to the (A)CS.

In the address field of your browser type:

```
192.168.160.10
```

Step 2: Log in.

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

Chapter 3 - Additional Features

Step 3: Click on the Edit Text File link.

Click on this link on the Link Panel or on the Configuration section of the Configuration and Administration page. (See [Figure 14: Configuration and Administration page](#). You can then pull up the appropriate file and edit it.

Step 4: Go to Configuration/Host Table.

Create/update the entry “ntphost.”

Step 5: Go to Configuration/Edit Text File.

Edit file and insert all parameter options needed.

Configuration for TS

vi Method

Same as for CAS.

Configuration for Dial-in Access

vi Method

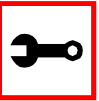
Same as for CAS.

Chapter 3 - Additional Features

PCMCIA



Warning! Although there are two PCMCIA sockets in the ACS, only one socket is currently supported. The bottom socket is the one available for use.



Note: This section applies only to the AlterPath ACS. The AlterPath CS has a single power supply.

Supported Cards

The ACS supports the 16-bit PC Cards. The 32-bit CardBus PC Cards are not supported. For an updated list of supported cards, please check the Cyclades Web site.

Tools for Configuring and Monitoring PCMCIA Devices

During the ACS boot, the `/etc/init.d/pcmcia` script loads the PCMCIA core drivers and the `cardmgr` daemon. The `cardmgr` daemon is responsible for monitoring PCMCIA sockets, loading client drivers when needed, and running user-level scripts in response to card insertions and removals.

lsmod This command shows the modules loaded for the PCMCIA devices

cardctl This command can be used to check the status of a socket, or to see how it is configured. Just type `cardctl` to see the syntax of the command. `cardctl config` displays the card configuration. `cardctl ident` can be used to get card identification information. `cardctl eject` stops the application and unloads the client driver, and `cardctl insert` re-loads the driver and re-starts the application.

Chapter 3 - Additional Features



Note: *cardctl suspend*, *cardctl resume* and *cardctl reset* are not supported.

Ejecting Cards

You can insert the card anytime, and the drivers should be loaded automatically. But you will need to run “*cardctl eject*” before ejecting the card to stop the application using the card. Otherwise the ACS may hang during the card removal.



Note: Due to a known problem in the current release, the I/O ports used by the card cannot be re-used after card re-insertion. In each card insertion, the card gets a different I/O port. This limits the number of times the card can be ejected and inserted. When all the I/O ports known by the card are used, the “RequestIO: No more items” message is displayed, and the only way to reset the I/O port usage is to reboot the system.

PCMCIA Network Configuration

The onboard Ethernet device has the *eth0* name. When the PCMCIA Ethernet card or wireless LAN card is detected, it will receive the *eth1* name.

cardmgr will read the network settings from the */etc/network/interfaces* and assign an IP to *eth1*.



Note: Before changing the */etc/network/interfaces* file, unload the network client driver using *cardctl eject*.

The factory default */etc/network/interfaces* has the following lines:

Chapter 3 - Additional Features

```
# auto eth1
# iface eth1 inet static
#     address 192.168.0.42
#     network 192.168.0.0
#     netmask 255.255.255.0
#     broadcast 192.168.0.255
#     gateway 192.168.0.1
```

Remove the # in the beginning of the line, and change the IPs to suit your network configuration. For instance, you may want the following configuration:

```
auto eth1
iface eth1 inet static
    address 192.168.162.10
    network 192.168.162.0
    netmask 255.255.255.0
    broadcast 192.168.162.255
    gateway 192.168.162.1
```

Don't forget to run *saveconf* to save this configuration in the flash, so that it can be restored in the next boot. Run *cardctl insert* to load the network drivers with the new configuration.



Note: Do not use *ifconfig* to change the network settings for the PCMCIA device. Otherwise, you may be unable to unload the network driver during *cardctl eject* and the ACS may hang. The correct way is to change the */etc/network/interfaces* file.

Chapter 3 - Additional Features

Modem PC Cards

The modem device gets the `/dev/ttySn` name, where *n* is the number of embedded serial devices plus 1. For instance, if the ACS has 32 onboard serial devices, the modem card becomes the `/dev/ttyS33`.

When a modem card is detected, *cardmgr* starts a script which loads *mgetty* for the modem device automatically.

Wireless LAN PC Cards

The configuration of the driver is done in the following file:

```
/lib/modules/2.4.17_mv121-linuxplanet/pcmcia-config/wireless.opts
```

For instance, to configure the network name as *MyPrivateNet*, and the Web encryption key as *secul*, the following settings could be added to the default `"*, *, *, *)"` entry :

```
*, *, *, *)  
  
INFO="This is a test"  
  
ESSID="MyPrivateNet"  
  
KEY="s:secul"
```



Note: The "s:" prefix in the KEY line indicates that the key is an ASCII string, as opposed to hex digits. Five characters or ten digits could be entered for WEP 40-bit and 13 characters or 26 digits could be entered for WEP 128-bit.

There is a generic sample in the end of the *wireless.opts* file that explains all possible settings. For more details in wireless configuration, search for *manpage iwconfig* on the Internet. The parameters in *wireless.opts* are used by the *iwconfig* utility. After changing any of the parameters, run *cardctl eject* followed by *cardctl insert* to load the new settings. Also, run *saveconf* to save the new settings to flash.

Chapter 3 - Additional Features

iwconfig eth1 shows the basic wireless parameters set in *eth1*. *iwlist* allows to list frequencies, bit-rates, encryption, etc. The usage is:

```
iwlist eth1 frequency
iwlist eth1 channel
iwlist eth1 ap
iwlist eth1 accesspoints
iwlist eth1 bitrate
iwlist eth1 rate
iwlist eth1 encryption
iwlist eth1 key
iwlist eth1 power
iwlist eth1 txpower
iwlist eth1 retry
```

Chapter 3 - Additional Features

Ports Configured for Dial-in Access

The AlterPath Console Server can be configured to accommodate out-of-band management. Ports can be configured on the AlterPath Console Server to allow a modem user to access the LAN. Radius authentication is used in this example and ppp is chosen as the protocol on the serial (dial-up) lines. Cyclades recommends that a maximum of two ports be configured for this option.

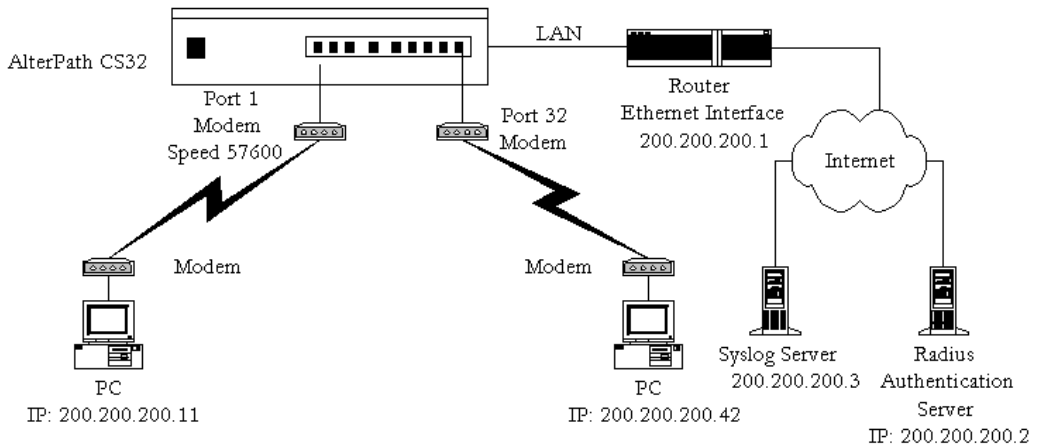


Figure 28: Ports configured for Dial-in Access

In addition to the parameters which are common to all setups, and which appear in [Appendix C - The pslave Configuration File](#), you may also configure additional parameters if you wish to configure some ports for Dial-in Access. These are also listed in the same section under [Dial-in Access Parameters](#). After configuring the desired parameters, execute the command `signal_ras hup` to activate the changes. At this point, the configuration should be tested. A step-by-step check list follows:



Note: If you add a user through the Web browser, the user does not actually get added to the list of users allowed to access the actual (A)CS unit.

Chapter 3 - Additional Features

Step 1: Create a new user.

Since RADIUS authentication was chosen, create a new user on the RADIUS authentication server called *test* and provide them with the password *test*.

Step 2: Confirm that the RADIUS server is reachable.

From the console, ping 200.200.200.2 to make sure the RADIUS authentication server is reachable.

Step 3: Confirm physical connections.

Make sure that the physical connection between the AlterPath Console Server and the modems is correct. The modem cable provided with the product should be used. Please see [Appendix B - Cabling, Hardware, and Electrical Specifications](#) for pinout diagrams.

Step 4: Confirm modem settings.

The AlterPath Console Server has been set for communication at 57600 bps, 8N1. The modems should be programmed to operate at the same speed on the DTE interface.

Step 5: Confirm routing.

Also make sure that the computer is configured to route console data to the serial console port.

Step 6: Perform a test dial-in.

Try to dial in to the AlterPath Console Server from a remote computer using the username and password configured in step one. The computer dialing in must be configured to receive its IP address from the remote access server (the AlterPath Console Server in this case) and to use PAP authentication.

Step 7: Activate changes.

Now continue on to [Task 5: Activate the changes](#) through [Task 8: Reboot the AlterPath Console Server](#) listed in Chapter 2 - Installation and Configuration.

Chapter 3 - Additional Features

Ports Configured as Terminal Servers

The AlterPath Console Server provides features for out-of-band management via the configuration of terminal ports. All ports can be configured as terminal ports. This allows a terminal user to access a server on the LAN. The terminal can be either a dumb terminal or a terminal emulation program on a PC.

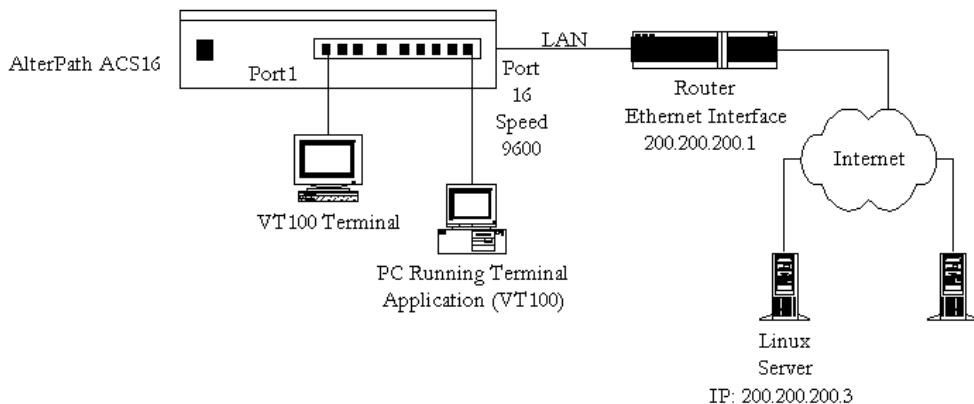


Figure 29: Terminal Server diagram

In addition to the parameters which are common to all setups, and which are listed in [Appendix C - The pslave Configuration File](#), you may also configure additional parameters for the Terminal Server port profile. They are listed in the same chapter under [TS Parameters](#).

Chapter 3 - Additional Features

TS Setup Scenario

No authentication is used in the example shown in [Figure 29: Terminal Server diagram](#) and rlogin is chosen as the protocol. After configuring the desired parameters, execute the command `signal_ras` to activate the configuration changes. At this point, the configuration should be tested. A step-by-step check list follows:

Step 1: Create a new user.

Since authentication was set to none, the AlterPath Console Server will not authenticate the user. However, the Linux Server receiving the connection will. Create a new user on the server called `test` and provide him with the password `test`.

Step 2: Confirm that the server is reachable.

From the console, ping 200.200.200.3 to make sure the server is reachable.

Step 3: Check physical connections.

Make sure that the physical connection between the AlterPath Console Server and the terminals is correct. A cross cable (not the modem cable provided with the product) should be used. Please see the [Appendix B - Cabling, Hardware, and Electrical Specifications](#) for pin-out diagrams.

Step 4: Confirm that terminals are set to same parameters as the (A)CTS.

The AlterPath Console Server has been set for communication at 9600 bps, 8N1. The terminals must also be configured with the same parameters.

Step 5: Log onto server with new username and password.

From a terminal connected to the AlterPath Console Server, try to login to the server using the username and password configured in step one.

Step 6: Activate changes.

Now continue on to [Task 5: Activate the changes](#) through [Task 8: Reboot the AlterPath Console Server](#) listed in [Chapter 2 - Installation and Configuration](#).

Chapter 3 - Additional Features

TS Setup Wizard

The Wizard can be used to configure TS-specific parameters. (TSO stands for “TS Other”- other parameters specific to the TS profile):

Step 1: At the command line interface type the following:

```
wiz --tso
```

Screen 1:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
Set to defaults? (y/n) [N]:
```

Screen 2:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.HOST - The IP address of the host to which the terminals will connect.

```
all.host[200.200.200.3] :
```

Chapter 3 - Additional Features

ALL.TERM - This parameter defines the terminal type assumed when performing rlogin or telnet to other hosts.

```
all.term[vt100] :
```

Screen 3:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

CONF.LOCALLOGINS - This parameter is only necessary when authentication is being performed for a port. When set to 1, it is possible to log into the system directly by placing a '!' before users' login name, then using their normal password. This is useful if the Radius authentication server is down.

```
conf.locallogins[0] :
```

Screen 4:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Your current configuration parameters are:

(The ones with the '#' means it's not activated.)

Chapter 3 - Additional Features

```
all.host : 200.200.200.3
all.term : vt100
conf.locallogins : 0
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

If you type 'N'

Type 'c' to go back and CORRECT these parameters
or 'q' to QUIT :

Typing 'c' repeats the application, typing 'q' exits the entire wiz application

If you type 'Y'

Type 'c' to CONTINUE to set these parameters for
specific ports or 'q' to QUIT :

Typing 'c' leads to Screen 5, typing 'q' leads to Screen 6.

Screen 5:

```
*****
***** C O N F I G U R A T I O N   W I Z A R D *****
*****
You have 8 available ports on this system.
Type 'q' to quit, a valid port number[1-8], or anything
else to refresh :
```



Tip. The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 6.

Chapter 3 - Additional Features

Screen 6:

```
*****  
***** C O N F I G U R A T I O N   W I Z A R D *****  
*****
```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

Serial Settings

This feature controls the speed, data size, parity, and stop bits of all ports. It also sets the flow control to hardware, software, or none; the DCD signal; and tty settings after a socket connection to that serial port is established.

Parameters Involved and Passed Values

Terminal Settings involve the following parameters (the first four are physical parameters):

<i>all.speed</i>	The speed for all ports. Example value: <i>9600</i> .
<i>all.datasize</i>	The data size for all ports. Example value: <i>8</i> .
<i>all.stopbits</i>	The number of stop bits for all ports. Example value: <i>1</i> .
<i>all.parity</i>	The parity for all ports. Example value: <i>none</i> .

Chapter 3 - Additional Features

- all.flow* This sets the flow control to hardware, software, or none.
Example value: *hard*.
- all.dcd* DCD signal (sets the tty parameter CLOCAL). Valid values are 0 or 1. In a socket session, if *all.dcd=0*, a connection request (telnet or ssh) will be accepted regardless of the DCD signal and the connection will not be closed if the DCD signal is set to DOWN. In a socket connection, if *all.dcd=1* a connection request will be accepted only if the DCD signal is UP and the connection (telnet or ssh) will be closed if the DCD signal is set to DOWN. Example value: *0*.
- all.sttycmd (for CAS only)* Tty settings after a socket connection to that serial port is established. The tty is programmed to work as a CAS configuration and this user-specific configuration is applied over that serial port. Parameters must be separated by a space. The following example sets *igncr*, which tells the terminal not to ignore the carriage-return on input, *-onlcr* do not map newline character to a carriage return/newline character sequence on output, *opost* post-process output, *-icrnl* do not map carriage-return to a newline character on input.
- ```
all.sttyCmd -igncr -onlcr opost -icrnl
```
- Example value: *commented*.
- DTR\_reset (for CAS only)* This value specifies how long (in milliseconds) a DTR signal will be turned off before it is turned back on again. If set to 0, this parameter will NOT be active. This may be dangerous if a user were to connect to a port that a previous user was on but had lost the session after a timeout. The user may directly connect into the previous user's shell. A minimum of 100ms is required otherwise it is assumed.

# Chapter 3 - Additional Features

---

---

## Configuration for CAS

### Browser Method

**Step 1: Point your browser to the (A)CS.**

In the address field of your browser type:

```
192.168.160.10
```

**Step 2: Log in.**

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

**Step 3: Select the Serial Ports link.**

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

**Step 4: Select port(s).**

On the Port Selection page, choose all ports or an individual port to configure, from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

**Step 5: Scroll down to the Physical section.**

You can change the settings for Speed, Data Size, Stop Bit, Parity, Flow Control, and DCD-sensitivity here.

### Wizard Method

**Step 1: Bring up the wizard.**

At the command prompt, type the following to bring up the CAS Terminal Settings custom wizard:

```
wiz --sset cas
```

Screen 1 will appear.



# Chapter 3 - Additional Features

---

## *Screen 1:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

Set to defaults? (y/n) [N]:
```

## *Screen 2:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.SPEED - The data speed in bits per second (bps) of all ports.

```
all.speed[9600] :
```

ALL.DATASIZE - The data size in bits per character of all ports.

```
all.datasize[8] :
```

# Chapter 3 - Additional Features

---

---

## Screen 3:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.STOPBITS - The number of stop bits for all ports.

all.stopbits[1] :

ALL.PARITY - The parity for all ports.  
(e.g. none, odd, even)

all.parity[none] :

## Screen 4:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.FLOW - This sets the flow control to hardware, software, or none. (e.g. hard, soft, none)

# Chapter 3 - Additional Features

---

all.flow[none] :

ALL.DCD - DCD signal (sets the tty parameter CLOCAL). Valid values are 0 or 1. In a socket session, if all.dcd=0, a connection request (telnet or ssh) will be accepted regardless of the DCD signal and the connection will not be closed if the DCD signal is set to DOWN. In a socket connection, if all.dcd=1 a connection request will be accepted only if the DCD signal is UP and the connection (telnet or ssh) will be closed if the DCD signal is set to DOWN.

all.dcd[0] :

## *Screen 5:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.DTR\_RESET - This parameter specifies how long (in milliseconds) a DTR signal will be turned off before it is turned on again. If set to 0, this parameter will NOT be active. This may be dangerous when a user connects to a port that a previous user was on but had lost the session after a timeout. The user may directly connect into the previous user's shell. A minimum of 100ms is required.

all.DTR\_reset[100] :

# Chapter 3 - Additional Features

---

---

ALL.STTYCMD - Tty settings after a socket connection to that serial port is established. The tty is programmed to work as a CAS profile and this user specific configuration is applied over that serial port. Parameters must be separated by space.(e.g. all.sttyCmd -igncr -onlcr opost -icrnl) -igncr tells the terminal not to ignore the carriage-return on input, -onlcr means do not map newline character to a carriage return/newline character sequence on output, opost represents post-process output, -icrnl means do not map carriage-return to a newline character on input.

```
all.sttyCmd[#] :
```

## *Screen 6:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

```
Your current configuration parameters are:
(The ones with the '#' means it's not activated.)
```

```
all.speed : 9600
all.datasize : 8
all.stopbits : 1
all.parity : none
all.flow : none
all.dcd : 0
all.DTR_reset : 100
all.sttyCmd : #
```

```
Are these configuration(s) all correct (Y)es or (N)o [N] :
```

## *If you type 'N'*

Type 'c' to go back and CORRECT these parameters  
or 'q' to QUIT :

*Typing 'c' repeats the application, typing 'q' exits the entire wiz application*

# Chapter 3 - Additional Features

---

## *If you type 'Y'*

Type 'c' to CONTINUE to set these parameters for specific ports or 'q' to QUIT :

*Typing 'c' leads to Screen 7, typing 'q' leads to Screen 8.*

### *Screen 7:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

You have 8 available ports on this system.
```

Type 'q' to quit, a valid port number[1-8], or anything else to refresh :



**Note:** The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 8.

### *Screen 8:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

# Chapter 3 - Additional Features

---

---

## CLI Method

To configure certain parameters for a specific serial port.

### Step 1: Bring up the CLI.

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

### Step 2: Type the following after the CLI prompt.

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

To configure speed:

```
configure line <serial port number> speed <number>
```

To configure datasize:

```
configure line <serial port number> datasize <number>
```

To configure stopbits:

```
configure line <serial port number> stopbits <number>
```

To configure parity:

```
configure line <serial port number> parity <string>
```



**Tip.** You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> speed
<number> datasize <number> stopbits <number> parity
<string>
```

# Chapter 3 - Additional Features

---

**Step 3:** To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

**Step 4:** To activate your new configurations, type:

```
signal_ras hup
```

## Configuration for TS

Browser Method

See the browser method for the CAS, earlier in this section.

Wizard Method

**Step 1:** Bring up the wizard.

At the command prompt, type the following to bring up the TS Terminal Settings custom wizard:

```
wiz --sset ts
```



**Note:** Screens 1- 4 are the same as those of the previous wizard for sset cas, thus, they are omitted here. The only difference between this feature and the CAS wizard is the parameter sttyCmd. In the TS configuration, sttyCmd is not requested.

**Screen 5:**

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Your current configuration parameters are:  
(The ones with the '#' means it's not activated.)

```
all.speed : 9600
all.datasize : 8
all.stopbits : 1
all.parity : none
```

# Chapter 3 - Additional Features

---

---

```
all.flow : none
all.dcd : 0
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

## *If you type 'N'*

Type 'c' to go back and CORRECT these parameters  
or 'q' to QUIT :

*Typing 'c' repeats the application, typing 'q' exits the entire wiz application*

## *If you type 'Y'*

Type 'c' to CONTINUE to set these parameters for specific ports or  
'q' to QUIT :

*Typing 'c' leads to Screen 6, typing 'q' leads to Screen 7.*

## *Screen 6:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

You have 8 available ports on this system.
Type 'q' to quit, a valid port number[1-8], or anything
else to refresh :
```



**Note:** The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 7.

## *Screen 7:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than



# Chapter 3 - Additional Features

---

one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

## CLI Method

To configure certain parameters for a specific serial port.

**Step 1: Bring up the CLI.**

At the command prompt, type the following to bring up the CLI.

```
config
```

This will show the CLI prompt

```
config@hostname>>
```

**Step 2: Type the following after the CLI prompt.**

To activate the serial port. <string> should be “ttyS<serial port number>”:

```
configure line <serial port number> tty <string>
```

**To configure speed:**

```
configure line <serial port number> speed <number>
```

**To configure datasize:**

```
configure line <serial port number> datasize <number>
```

**To configure stopbits:**

```
configure line <serial port number> stopbits <number>
```

**To configure parity:**

# Chapter 3 - Additional Features

---

---

```
configure line <serial port number> parity <string>
```



**Tip.** You can configure all the parameters for a serial port in one line.

```
configure line <serial port number> tty <string> speed
<number> datasize <number> stopbits <number> parity
<string>
```

**Step 3:** To exit the CLI.

Type *exit* or *quit* after the CLI prompt.

**Step 4:** To activate your new configurations, type:

```
signal_ras hup
```

## Configuration for Dial-in Access

The parameters are the same as before.

## Session Sniffing

### Versions 1.3.2 and earlier

The AlterPath Console Server allows a maximum of two connections to each serial port, as follows:

- One common session: user can execute read and write commands to the tty port. Session can be established by a regular user or by an administrator.
- One sniffer session: user can execute only read commands, in order to monitor what is going on in the other (main) session. Session can only be established by an administrator, defined by the parameter `all.admin_users` or `sN.admin_users` in the file `pslave.conf` (exception: authentication none - anyone can open a sniffer).

# Chapter 3 - Additional Features

---

---

The first connection always opens a common session. After the second connection has been established and the user is authenticated, the AlterPath Console Server shows the following menu to the administrator user:

```

*
* * * ttySN is being used by (<user_name>) !!!
*
1 - Assume the main session
2 - Initiate a sniff session
3 - Quit
Enter your option:

```

If the second user is not an administrator, his connection is automatically refused. This description is valid for all of the available protocols (socket\_server, socket\_ssh or raw\_data).

## Versions 1.3.3 and later

Users will be able to open more than one common and sniff session at the same port. For this purpose, the following configuration items are available in the file pslave.conf:

- `all.multiple_sessions`: valid for all the serial ports; must be “yes” or “no.” The default value is “no.”
- `sN.multiple_sessions`: valid only for port N; must be “yes” or “no.” If it is not defined, it will assume the value of `all.multiple_sessions`.
- `all.escape_char`: valid for all the serial ports; this parameter will be used to present the menus below to the user. Only characters from ‘^a’ to ‘^z’ (i.e., CTRL-A to CTRL-Z) will be accepted. The default value is ‘^z’ (CTRL-Z).
- `sN.escape_char`: valid only for port N; this parameter will be used to present the menus below to the user. Only characters from ‘^a’ to ‘^z’ (i.e. CTRL-A to CTRL-Z) will be accepted. If it is not defined, it will assume the value of `all.escape_char`.

# Chapter 3 - Additional Features

---

---

When no multiple sessions are allowed for one port, the behavior of the AlterPath Console Server when someone connects to it will be as described for version 1.3.2 and earlier. Otherwise, it will be as follows:

1. The first user to connect to the port will open a common session.
2. From the second connection on, only admin users will be allowed to connect to that port. The AlterPath Console Server will send the following menu to these administrators (defined by the parameter `all.admin_users` or `sN.admin_users` in the file `pslave.conf`):

```

*
* * * ttySN is being used by (<first_user_name>) !!!
*
1 - Initiate a regular session
2 - Initiate a sniff session
3 - Send messages to another user
4 - Kill session(s)
5 - Quit
Enter your option:

```

If the user selects *1 - Initiate a regular session*, s/he will share that serial port with the users that were previously connected. S/he will read everything that is received by the serial port, and will also be able to write to it.

If the user selects *2 - Initiate a sniff session*, s/he will start reading everything that is sent and/or received by the serial port, according to the parameter `all.sniff_mode` or `sN.sniff_mode` (that can be in, out or i/o).

When the user selects *3 - Send messages to another user*, the AlterPath Console Server will send the user's messages to all the sessions, but not to the tty port. Everyone connected to that port will see all the "conversation" that's going on, as if they were physically in front of the console in the same room. These messages will be formatted as:

# Chapter 3 - Additional Features

---

[Message from user/PID] <<message text goes here>> by the (A)CS

To inform the AlterPath Console Server that the message is to be sent to the serial port or not, the user will have to use the menu.

If the administrator chooses the option *4 - Kill session(s)*, the AlterPath Console Server will show him/her a list of the pairs PID/user\_name, and s/he will be able to select one session typing its PID, or “all” to kill all the sessions.

*Option 5 - Quit* will close the current session and the TCP connection.

Only for the administrator users: typing `all.escape_char` or `sN.escape_char` from the normal or sniff session or “send message mode” will make the (A)CS show the previous menu. If this parameter is not set in `pslave.conf`, or it contains an invalid value, the regular sessions will not be allowed to return to the menu, and the sniffer sessions will be able to do it typing `<CTRL-Z>`. In addition, the regular session will only be allowed to see the menu if the protocol used is “`socket_server`” or “`socket_ssh`”.

## Parameters Involved and Passed Values

Sniffing involves the following parameters:

### *all.admin\_users*

This parameter determines which users can open a sniff session, which is where other users connected to the very same port can see everything that a previously-connected user is doing. The other users connected to the very same port can also cancel the first user’s session (and take over). If `all.multiple_sessions` (seen below) is configured as *no*, only two users can connect to the same port simultaneously. If `all.multiple_sessions` is configured as *yes*, more simultaneous users can sniff the session or have read and/or write permission. When users want access per port to be controlled by administrators, this parameter is obligatory and `authtype` must not be *none*. This parameter can determine who can open a sniff session or cancel a previous session. User groups (defined with the parameter `conf.group`) can be used in combination with user names in the parameter list. Example values: `peter, john, user_group`.

# Chapter 3 - Additional Features

---

---

- all.sniff\_mode* This parameter determines what other users connected to the very same port (see parameter `admin_users` below) can see of the session of the first connected user (main session): *in* shows data written to the port, *out* shows data received from the port, and *i/o* shows both streams. The second and later sessions are called sniff sessions and this feature is activated whenever the protocol parameter is set to `socket_ssh` or `socket_server`. Example value: `out`.
- all.escape\_char* This parameter determines which character must be typed to make the session enter "menu mode". The possible values are <CTRL-a> to <CTRL-z>. Represent the CTRL with '^'. This parameter is only valid when the port protocol is `socket_server` or `socket_ssh`. Default value is '^z'.
- all.multiple\_sessions* Valid for all serial ports. Must be "yes" or "no." If it is not defined, the default will be "no". Example value: `yes`.

## Configuration for CAS

### vi Method

Only the file `/etc/portslave/pslave.conf` has to be changed.

### Browser Method

To configure Session Sniffing with your browser:

**Step 1: Point your browser to the (A)CS.**

In the address field of your browser type:

```
192.168.160.10
```

**Step 2: Log in.**

Log in as root, `pwd` is `tslinux`. This will take you to the Configuration and Administration page.

**Step 3: Select the Serial Ports link.**

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

# Chapter 3 - Additional Features

---

Step 4: Select port(s).

On the Port Selection page, choose all ports or an individual port to configure, from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

Step 5: Scroll down to the Sniff Session section.

You can configure the appropriate values here.

| Sniff session                   |                                                               |
|---------------------------------|---------------------------------------------------------------|
| Sniff Session Mode:             | Output                                                        |
| Administrative Users:           |                                                               |
| Escape char from sniff mode:    |                                                               |
| Allows multiple sniff sessions: | <input type="radio"/> yes <input checked="" type="radio"/> no |

*Figure 30: Sniff Session section of the Serial Port Configuration page*

## Wizard Method

Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Sniffing custom wizard:

```
wiz --snf
```

*Screen 1:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

Set to defaults? (y/n) [N]:
```

# Chapter 3 - Additional Features

---

---

## Screen 2:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.ADMIN\_USERS - This parameter determines which users can open a sniff session, which is where other users connected to the very same port can see everything that the first user is doing. The other users connected to the very same port can also cancel the first user's session (and take over). If the parameter, all.multiple\_sessions, is configured as 'no', then only two users can connect to the same port simultaneously. If it is configured as 'yes', more simultaneous users can sniff the session or have read/write permissions.

(Please see details in Session Sniffing in Chapter 3 of the system's manual.)

all.admin\_users[#] :

ALL.SNIFF\_MODE - This parameter determines what other users connected to the very same port can see of the session of the first connected user (main session). The second session is called a sniff session and this feature is activated whenever the protocol is set to socket\_ssh or socket\_server.

(e.g. in -shows data written to the port, out -shows data received from the port, i/o -shows both streams.)



# Chapter 3 - Additional Features

---

all.sniff\_mode[out] :

## *Screen 3:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.ESCAPE\_CHAR - This parameter determines which character must be typed to make the session enter into "menu mode." The possible values are <CTRL-a> to <CTRL-z>, and this is only valid when the port protocol is socket\_server or socket\_ssh. Represent the CTRL character with '^'. Default value is ^z.

all.escape\_char[^z] :

ALL.MULTIPLE\_SESSIONS - Allow users to open more than one

# Chapter 3 - Additional Features

---

---

common and sniff sessions on the same port. The parameter must be a 'yes' or a 'no' to open. Default is set to 'no'.

```
all.multiple_sessions[no] :
```

## ***Screen 4:***

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Your current configuration parameters are:  
(The ones with the '#' means it's not activated.)

```
all.admin_users : #
all.sniff_mode : out
all.escape_char : ^z
all.multiple_sessions : no
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

### ***If you type 'N'***

Type 'c' to go back and CORRECT these parameters  
or 'q' to QUIT :

***Typing 'c' repeats the application, typing 'q' exits the entire wiz application***

### ***If you type 'Y'***

Type 'c' to CONTINUE to set these parameters for  
specific ports or 'q' to QUIT :

***Typing 'c' leads to Screen 5, typing 'q' leads to Screen 6.***

# Chapter 3 - Additional Features

---

## Screen 5:

```

***** C O N F I G U R A T I O N W I Z A R D *****

You have 8 available ports on this system.
Type 'q' to quit, a valid port number[1-8], or anything
else to refresh :
```



**NOTE:** The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 6.

## Screen 6:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

# Chapter 3 - Additional Features

---

---

## SNMP

Short for Simple Network Management Protocol: a set of protocols for managing complex networks. The first versions of SNMP were developed in the early 80s. SNMP works by sending messages, called protocol data units (PDUs), to different parts of a network. SNMP-compliant devices, called agents, store data about themselves in Management Information Bases (MIBs) and return this data to the SNMP requesters.

The (A)CS uses the net-snmp package (<http://www.net-snmp.org>).

The net-snmp supports snmp version 1, 2 and 3. You can configure the `/etc/snmp/snmpd.conf` file as indicated later in this section.

### 1. Snmp version 1

- RFC1155 - SMI for the official MIB tree
- RFC1213 - MIB-II

### 2. Snmp version 2

- RFC2578 - Structure of Management Information Version 2 (SMIv2)
- RFC2579 - Textual Conventions for SMIv2
- RFC2580 - Conformance Statements for SMIv2

### 3. Snmp version 3

- RFC2570 - Introduction to Version 3 of the Internet-standard Network Management Framework
- RFC2571 - An Architecture for Describing SNMP Management Frameworks
- RFC2572 - Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
- RFC2573 - SNMP Applications
- RFC2574 - User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)

# Chapter 3 - Additional Features

---

- RFC2575 - View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)
  - RFC2576 - Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework
4. Private UCD SNMP mib extensions (enterprises.2021)
    - Information about memory utilization (/proc/meminfo)
    - Information about system status (vmstat)
    - Information about net-snmp packet
  5. Private Cyclades Vendor MIB ( enterprises.2925 )
    - Cyclades ACSxx Remote Management Object Tree (cyclades.4). This MIB permits you to get informations about the product, to read/write some configuration items and to do some administration commands. (For more details see the cyclades.mib file.)

## Configuration for CAS

vi Method

Files to be changed:

/etc/snmp/snmpd.conf

This file has information about configuring for SNMP.

Browser Method

To configure SNMP with your browser:

**Step 1: Point your browser to the (A)CS.**

In the address field of your browser type:

192.168.160.10

**Step 2: Log in.**

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

# Chapter 3 - Additional Features

---

---

Step 3: Click on the Edit Text File link.

Click on this link on the Link Panel or on the Configuration section of the Configuration and Administration page. (See [Figure 14: Configuration and Administration page](#). You can then pull up the appropriate file and edit it.

## Configuration for TS

vi Method

Same as for CAS.

## Configuration for Dial-in Access

vi Method

Same as for CAS.

## Syslog

The syslog-ng daemon provides a modern treatment to system messages. Its basic function is to read and log messages to the system console, log files, other machines (remote syslog servers) and/or users as specified by its configuration file. In addition, syslog-ng is able to filter messages based on the contents of them and to perform an action (e.g. to send an e-mail or pager message). In order to access these functions, the syslog-ng.conf file needs some specific configuration.

The configuration file (default: syslog-ng.conf) is read at startup and is reread after receipt of a hangup (HUP) signal. When reloading the configuration file, all destination files are closed and reopened as appropriate.

The syslog-ng reads from sources (files, TCP/UDP connections, syslogd clients), filters the messages and takes an action (writes in files, sends snmptrap, pager, e-mail or syslogs to remote servers).

# Chapter 3 - Additional Features

---

There are five tasks required for configuring syslog-ng:

- Task 1: Define Global Options.
- Task 2: Define Sources.
- Task 3: Define Filters.
- Task 4: Define Actions (Destinations).
- Task 5: Connect all of the above.

The five tasks are explained in the following section [“Syslog-ng and its Configuration” on page 188](#).

## Port Slave Parameters Involved with syslog-ng

|                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>conffacility</i>         | This value (0-7) is the Local facility sent to the syslog-ng from PortSlave.                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <i>confDB_facility</i>      | This value (0-7) is the Local facility sent to the syslog-ng with data when <code>syslog_buffering</code> and/or <code>alarm</code> is active. When nonzero, the contents of the data buffer are sent to the syslog-ng every time a quantity of data equal to this parameter is collected. The syslog level for data buffering is hard coded to level five (notice) and facility <code>conf.DB_facility</code> . The file <code>/etc/syslog-ng/syslog-ng.conf</code> should be set accordingly for the syslog-ng to take some action. Example value: 0. |
| <i>all.syslog_buffering</i> | When non zero, the contents of the data buffer are sent to the syslog-ng every time a quantity of data equal to this parameter is collected. The syslog message is sent to syslog-ng with NOTICE level and <code>LOCAL[0+conf.DB_facility]</code> facility.                                                                                                                                                                                                                                                                                             |

## Configuration for CAS

vi Method

File to be changed: `pslave.conf`. The parameters are the same.

# Chapter 3 - Additional Features

---

---

## Browser Method

To configure syslog via your Web browser:

**Step 1: Point your browser to 192.168.130.10.**

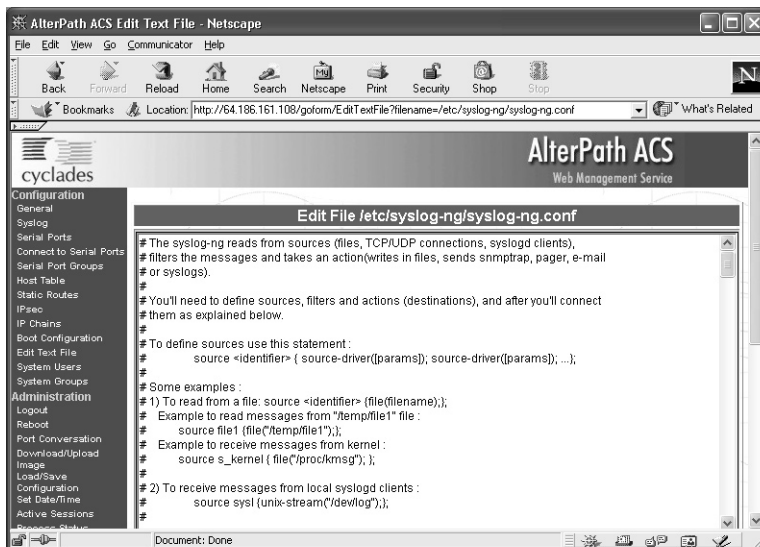
Enter the (A)CS's IP address in your browser's address field.

**Step 2: Log in.**

Enter *root* as the username and *tslinux* as the password. This will take you to the Configuration and Administration Menu Page: [“Configuration & Administration Menu page” on page 34.](#)

**Step 3: Click Syslog on the Configuration section.**

Select the Syslog link. The following page will appear, giving information for configuring syslog:



*Figure 31: Syslog page 1*

**Step 4: Click the Edit Text File link on the Link Panel.**

Enter the filename and begin editing the file.



# Chapter 3 - Additional Features

---

## Wizard Method

### Step 1: Bring up the wizard.

At the command prompt, type the following to bring up the Syslog custom wizard:

```
wiz --sl
```

Screen 1 will appear.

### Screen 1:

```

***** C O N F I G U R A T I O N W I Z A R D *****

Set to defaults? (y/n) [N]:
```

### Screen 2:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

CONF.FACILITY - This value (0-7) is the Local facility sent to the syslog. The file /etc/syslog-ng/syslog-ng.conf contains a mapping between the facility number and the action.

(Please see the 'Syslog-ng Configuration to use with Syslog Buffering Feature' section under Generating Alarms in

# Chapter 3 - Additional Features

---

---

Chapter 3 the system's manual for the syslog-ng configuration file.)

```
conf.facility[7] :
```

CONF.DB\_FACILITY - This value (0-7) is the Local facility sent to the syslog with the data when syslog\_buffering is active. The file /etc/syslog-ng/syslog-ng.conf contains a mapping between the facility number and the action. (Please see the 'Syslog-ng Configuration to use with Syslog Buffering Feature' section under Generating Alarms in Chapter 3 the system's manual for the syslog-ng configuration file.)

```
conf.DB_facility[0] :
```

### *Screen 3:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Your current configuration parameters are:  
(The ones with the '#' means it's not activated.)

```
conf.facility : 7
conf.DB_facility : 0
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

### *If you type 'N'*

Type 'c' to go back and CORRECT these parameters  
or 'q' to QUIT :

*Typing 'c' repeats the application, typing 'q' exits the entire wiz application*

### *If you type 'Y'*

Type 'c' to CONTINUE to set these parameters for  
specific ports or 'q' to QUIT :

*Typing 'c' leads to Screen 4, typing 'q' leads to Screen 5.*

# Chapter 3 - Additional Features

---

## Screen 4:

```

***** C O N F I G U R A T I O N W I Z A R D *****

You have 8 available ports on this system.
Type 'q' to quit, a valid port number[1-8], or anything
else to refresh :
```



**NOTE:** The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 5.

## Screen 5:

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

## Configuration for TS

vi Method

Same as for CAS.

# Chapter 3 - Additional Features

---

---

## Configuration for Dial-in Access

vi Method

Same as for CAS.

## The Syslog Functions

This section shows the characteristics of the Alarm for Data Buffering that is implemented for all members of the AlterPath Console Server family. It is divided into three parts:

1. [Syslog-ng and its Configuration](#)
2. [Syslog-ng Configuration to use with Syslog Buffering Feature](#)
3. [Syslog-ng Configuration to use with Multiple Remote Syslog Servers](#)

### Syslog-ng and its Configuration

The five tasks previously mentioned are detailed below.

#### Task 1: Specify Global Options.

You can specify several global options to syslog-ng in the options statement:

```
options { opt1(params); opt2(params); ... };
```

where *optn* can be any of the following:

|                                                                          |                                                                                                                                |
|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <i>time_reopen(n)</i>                                                    | The time to wait before a dead connection is reestablished.                                                                    |
| <i>time_reap(n)</i>                                                      | The time to wait before an idle destination file is closed.                                                                    |
| <i>sync_freq(n)</i>                                                      | The number of lines buffered before written to file. (The file is synced when this number of messages has been written to it.) |
| <i>mark_freq(n)</i>                                                      | The number of seconds between two MARKS lines.                                                                                 |
| <i>log_fifo_size(n)</i>                                                  | The number of lines fitting to the output queue.                                                                               |
| <i>chain_hostname</i><br>(yes/no) or<br><i>long_hostname</i><br>(yes/no) | Enable/disable the chained hostname format.                                                                                    |

# Chapter 3 - Additional Features

---

|                                   |                                                                                                                                                                   |
|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>use_time_recvd</i><br>(yes/no) | Use the time a message is received instead of the one specified in the message.                                                                                   |
| <i>use_dns</i> (yes/no)           | Enable or disable DNS usage. syslog-ng blocks on DNS queries, so enabling DNS may lead to a Denial of Service attack.                                             |
| <i>gc_idle_threshold</i> (n)      | Sets the threshold value for the garbage collector, when syslog-ng is idle. GC phase starts when the number of allocated objects reach this number. Default: 100. |
| <i>gc_busy_threshold</i> (n)      | Sets the threshold value for the garbage collector. When syslog-ng is busy, GC phase starts.                                                                      |
| <i>create_dirs</i> (yes/no)       | Enable the creation of new directories.                                                                                                                           |
| <i>owner</i> (name)               | Set the owner of the created file to the one specified. Default: root.                                                                                            |
| <i>group</i> (name)               | Set the group of the created file to the one specified. Default: root.                                                                                            |
| <i>perm</i> (mask)                | Set the permission mask of the created file to the one specified. Default: 0600.                                                                                  |

## Task 2: Define sources.

To define sources use this statement:

```
source <identifier> { source-driver([params]); source
driver([params]); ...};
```

where:

|                      |                                                                                               |
|----------------------|-----------------------------------------------------------------------------------------------|
| <i>identifier</i>    | Has to uniquely identify this given source.                                                   |
| <i>source-driver</i> | Is a method of getting a given message.                                                       |
| <i>params</i>        | Each source-driver may take parameters. Some of them are required, some of them are optional. |

# Chapter 3 - Additional Features

---

---

The following source-drivers are available:

- a) internal()* Messages are generated internally in syslog-ng.
- b) unix-stream (filename [options])* They open the given AF\_UNIX socket, and start listening for messages.  
Options: owner(name), group(name), perm(mask) are equal global options
- and*
- unix-dgram (filename [options])* *keep-alive (yes/no)* - Selects whether to keep connections opened when syslog-ng is restarted. Can be used only with *unix\_stream*. Default: yes  
*max-connections(n)* - Limits the number of simultaneously opened connections. Can be used only with *unix\_stream*. Default: 10.
- c) tcp([options])* These drivers let you receive messages from the network, and as the name of the drivers show, you can use both TCP and UDP.
- and* None of *tcp()* and *udp()* drivers require positional parameters. By default they bind to 0.0.0.0:514, which means that syslog-ng will listen on all available interfaces.
- udp([options])* Options:  
*ip(<ip address>)* - The IP address to bind to. Default: 0.0.0.0.  
*port(<number>)* - UDP/TCP port used to listen messages. Default: 514.  
*max-connections(n)* - Limits the number of simultaneously opened connections. Default: 10.
- d) file(filename)* Opens the specified file and reads messages.
- e) pipe(filename)* Opens a named pipe with the specified name, and listens for messages. (You'll need to create the pipe using *mkfifo* command).

Some Examples of Defining Sources

1) To read from a file:

```
source <identifier> {file(filename)};
```

Example to read messages from "/temp/file1" file:

```
source file1 {file('/temp/file1')};
```

# Chapter 3 - Additional Features

---

Example to receive messages from the kernel:

```
source s_kernel { file('/proc/kmsg'); };
```

2) To receive messages from local syslogd clients:

```
source sysl { unix-stream('/dev/log'); };
```

3) To receive messages from remote syslogd clients:

```
source s_udp { udp(ip(<cliente ip>) port(<udp port>)); };
```

Example to listen to messages from all machines on UDP port 514:

```
source s_udp { udp(ip(0.0.0.0) port(514)); };
```

Example to listen to messages from one client (IP address=10.0.0.1) on UDP port 999:

```
source s_udp_10 { udp(ip(10.0.0.1) port(999)); };
```

## Task 3: Define filters.

To define filters use this statement:

```
filter <identifier> { expression; };
```

where:

- identifier*** Has to uniquely identify this given filter.
- expression*** Boolean expression using internal functions, which has to evaluate to true for the message to pass.

# Chapter 3 - Additional Features

---

---

The following internal functions are available:

- a) *facility(<facility code>)*           Selects messages based on their facility code.
- b) *level(<level code>)* or *priority(<level code>)*   Selects messages based on their priority.
- c) *program(<string>)*           Tries to match the <string> to the program name field of the log message.
- d) *host(<string>)*           Tries to match the <string> to the hostname field of the log message.
- e) *match(<string>)*           Tries to match the <string> to the message itself.

Some Examples of Defining Filters

1) To filter by facility:

```
filter f_facilty { facility(<facility name>); };
```

Examples:

```
filter f_daemon { facility(daemon); };
```

```
filter f_kern { facility(kern); };
```

```
filter f_debug { not facility(auth, authpriv, news, mail); };
```

2) To filter by level:

```
filter f_level { level(<level name>);};
```

Examples:

```
filter f_messages { level(info .. warn);}
```

```
filter f_emergency { level(emerg); };
```

```
filter f_alert { level(alert); };
```



# Chapter 3 - Additional Features

---

3) To filter by matching one string in the received message:

```
filter f_match { match('string'); };
```

**Example to filter by matching the string “named”:**

```
filter f_named { match('named'); };
```

4) To filter ALARM messages (note that the following three examples should be one line):

```
filter f_alarm { facility(local[0+<conf.DB_facility>]) and
level(info) and match('ALARM') and match('<your string>'); } ;
```

**Example to filter ALARM message with the string “kernel panic”:**

```
filter f_kpanic { facility(local[0+<conf.DB_facility>]) and
level(info) and match('ALARM') and match('kernel panic'); };
```

**Example to filter ALARM message with the string “root login”:**

```
filter f_root { facility(local[0+<conf.DB_facility>]) and
level(info) and match('ALARM') and match('root login'); };
```

5) To eliminate sshd debug messages:

```
filter f_sshd_debug { not program('sshd') or not level(debug); };
```

6) To filter the syslog\_buffering:

```
filter f_syslog_buf { facility(local[0+<conf.DB_facility>]) and
level(notice); };
```

## **Task 4: Define Actions.**

To define actions use this statement (note that the statement should be one line):

```
destination <identifier> { destination-driver([params]);
destination-driver([param]); ..};
```

# Chapter 3 - Additional Features

---

---

where:

|                           |                                                                                                |
|---------------------------|------------------------------------------------------------------------------------------------|
| <i>identifier</i>         | Has to uniquely identify this given destination.                                               |
| <i>destination driver</i> | Is a method of outputting a given message.                                                     |
| <i>params</i>             | Each destination-driver may take parameters. Some of them required, some of them are optional. |

The following destination drivers are available:

## *a) file(filename [options])*

This is one of the most important destination drivers in syslog-ng. It allows you to output log messages to the named file. The destination filename may include macros (by prefixing the macro name with a '\$' sign) which gets expanded when the message is written. Since the state of each created file must be tracked by syslog-ng, it consumes some memory for each file. If no new messages are written to a file within 60 seconds (controlled by the `time_reap` global option), it's closed, and its state is freed.

Available macros in filename expansion:

HOST - The name of the source host where the message originated from.

FACILITY - The name of the facility the message is tagged as coming from.

PRIORITY or LEVEL - The priority of the message.

PROGRAM - The name of the program the message was sent by.

YEAR, MONTH, DAY, HOUR, MIN, SEC - The year, month, day, hour, min, sec of the message was sent.

TAG - Equals FACILITY/LEVEL.

FULLHOST - The name of the source host and the source-driver:

<source-driver>@<hostname>

MSG or MESSAGE - The message received.

FULLDATE - The date of the message was sent.

Available options:

*log\_fifo\_size(number)* - The number of entries in the output file.

*sync\_freq(number)* - The file is synced when this number of messages has been written to it.

*owner(name), group(name), perm(mask)* - Equals global options.

*template("string")* - Syslog-ng writes the "string" in the file. You can use the MACROS in the string.

# Chapter 3 - Additional Features

---

*encrypt(yes/no)* - Encrypts the resulting file.

*compress(yes/no)* - Compresses the resulting file using zlib.

b) *pipe(filename [options])*

This driver sends messages to a named pipe. Available options:

*owner(name), group(name), perm(mask)* - Equals global options.

*template("string")* - Syslog-ng writes the "string" in the file. You can use the MACROS in the string.

c) *unix-stream(filename) and unix-dgram(filename)*

This driver sends messages to a UNIX socket in either SOCKET\_STREAM or SOCK\_DGRAM mode.

d) *udp("<ip address>" port(number);) and tcp("<ip address>" port(number);)*

This driver sends messages to another host (ip address/port) using either UDP or TCP protocol.

e) *usertty(<username>)*

This driver writes messages to the terminal of a logged-in username.

f) *program(<program name and arguments>)*

This driver fork()'s executes the given program with the arguments and sends messages down to the stdin of the child.

## Some Examples of Defining Actions

1) To send e-mail:

```
destination <ident> { pipe('/dev/cyc_alarm' template('sendmail
<pars>'))};
```

where *ident*: uniquely identifies this destination. Parameters:

|                                         |                  |
|-----------------------------------------|------------------|
| <i>-t &lt;name&gt;[,&lt;name&gt;]</i>   | To address       |
| <i>[-c &lt;name&gt;[,&lt;name&gt;]]</i> | CC address       |
| <i>[-b &lt;name&gt;[,&lt;name&gt;]]</i> | Bcc address      |
| <i>[-r &lt;name&gt;[,&lt;name&gt;]]</i> | Reply-to address |
| <i>-f &lt;name&gt;</i>                  | From address     |

# Chapter 3 - Additional Features

---

---

|                                                   |                       |
|---------------------------------------------------|-----------------------|
| <code>-s \<i>&lt;text&gt;</i>\</code>             | Subject               |
| <code>-m \<i>&lt;text message&gt;</i>\</code>     | Message               |
| <code>-h <i>&lt;IP address or name&gt;</i></code> | SMTP server           |
| <code>[<i>-p &lt;port&gt;</i>]</code>             | Port used. default:25 |

To mount the message, use this macro:

|                                                    |                                                                                                            |
|----------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| <code>\$FULLDATE</code>                            | The complete date when the message was sent.                                                               |
| <code>\$FACILITY</code>                            | The facility of the message.                                                                               |
| <code>\$PRIORITY</code> or<br><code>\$LEVEL</code> | The priority of the message.                                                                               |
| <code>\$PROGRAM</code>                             | The message was sent by this program (BUFFERING or SOCK).                                                  |
| <code>\$HOST</code>                                | The name of the source host.                                                                               |
| <code>\$FULLHOST</code>                            | The name of the source host and the source driver. Format:<br><code>&lt;source&gt;@&lt;hostname&gt;</code> |
| <code>\$MSG</code> or <code>\$MESSAGE</code>       | The message received.                                                                                      |

Example to send e-mail to `z@none.com` (SMTP's IP address 10.0.0.2) from the e-mail address `a@none.com` with subject "(A)CS-ALARM". The message will carry the current date, the host-name of this (A)CS and the message that was received from the source.

```
destination d_mail1 {
 pipe('/dev/cyc_alarm'
 template('sendmail -t z@none.com -f a@none.com -s \'(A)CS-ALARM\' \
 -m \'$FULLDATE $HOST $MSG\' -h 10.0.0.2'));
};
```

# Chapter 3 - Additional Features

---

## 2) To send to pager server (sms server):

```
destination <ident> {pipe(`/dev/cyc_alarm' template('sendsms
<pars>'))};
```

where ident: uniquely identify this destination

pars: -d <mobile phone number>

-m \`<message - max.size 160 characters>`\'

-u <username to login on sms server>

-p <port sms - default : 6701>

<server IP address or name>

**Example to send a pager to phone number 123 (Pager server at 10.0.0.1) with message carrying the current date, the hostname of this (A)CS and the message that was received from the source:**

```
destination d_pager {
pipe(`/dev/cyc_alarm'
template('sendsms -d 123 -m \'$FULLDATE $HOST $MSG\' 10.0.0.1'));
};
```

## 3) To send snmptrap:

```
destination <ident> {pipe(`/dev/cyc_alarm' template('snmptrap
<pars>'))};
```

where ident : uniquely identify this destination

pars : -v 1

<snmptrapd IP address>

-c public : community

\" : enterprise-oid

\" : agent/hostname



# Chapter 3 - Additional Features

---

6) To send a message to a remote syslogd server:

```
destination d_udp { udp("<remote IP address>" port(514)); };
```

Example to send syslogs to syslogd located at 10.0.0.1 :

```
destination d_udp1 { udp("10.0.0.1" port(514)); };
```

**Task 5: Connect all of the above.**

To connect the sources, filters, and actions, use the following statement. (Actions would be any message coming from one of the listed sources. A match for each of the filters is sent to the listed destinations.)

```
log { source(S1); source(S2); ...
filter(F1);filter(F2);...
destination(D1); destination(D2);...
};
```

where :

|           |                                                        |
|-----------|--------------------------------------------------------|
| <i>Sx</i> | Identifier of the sources defined before.              |
| <i>Fx</i> | Identifier of the filters defined before.              |
| <i>Dx</i> | Identifier of the actions/destinations defined before. |

Examples:

1) To send all messages received from local syslog clients to console:

```
log { source(sysl); destination(d_console);};
```

2) To send only messages with level alert and received from local syslog clients to all logged root user:

```
log { source(sysl); filter(f_alert); destination(d_userroot); };
```

# Chapter 3 - Additional Features

---

---

3) To write all messages with levels info, notice, or warning and received from syslog clients (local and remote) to `/var/log/messages` file:

```
log { source(sysl); source(s_udp); filter(f_messages); destination(d_messages); };
```

4) To send e-mail if message received from local syslog client has the string “kernel panic”:

```
log { source(sysl); filter(f_kpanic); destination(d_maill); };
```

5) To send e-mail and pager if message received from local syslog client has the string “root login”:

```
log { source(sysl); filter(f_root); destination(d_maill); destination(d_pager); };
```

6) To send messages with facility kernel and received from syslog clients (local and remote) to remote syslogd:

```
log { source(sysl); source(s_udp); filter(f_kern); destination(d_udp1); };
```

Syslog-ng Configuration to use with Syslog Buffering Feature

This configuration example uses the syslog buffering feature, and sends messages to the remote syslogd (10.0.0.1).

**Step 1: Configure `pslave.conf` parameters.**

In the `pslave.conf` file the parameters of the syslog buffering feature are configured as:

```
conf.DB_facility 1
all.syslog_buffering 100
```

**Step 2: Add lines to `syslog-ng.conf`.**

Add the following lines by vi or browser to the file:

```
local syslog clients
source src { unix-stream("/dev/log"); };
destination d_buffering { udp("10.0.0.1"); };
```



# Chapter 3 - Additional Features

---

```
filter f_buffering { facility(local1) and level(notice); };

send only syslog_buffering messages to remote server

log { source(src); filter(f_buffering); destination(d_buffering); };
```

Syslog-ng Configuration to use with Multiple Remote Syslog Servers

This configuration example is used with multiple remote syslog servers.

**Step 1: Configure pslave.conf parameters.**

In the pslave.conf file the facility parameter is configured as:

```
conf.facility 1
```

**Step 2: Add lines to syslog-ng.conf.**

The syslog-ng.conf file needs these lines:

```
local syslog clients
source src { unix-stream("/dev/log"); };

remote server 1 - IP address 10.0.0.1 port default
destination d_udp1 { udp("10.0.0.1"); };

remote server 2 - IP address 10.0.0.2 port 1999
destination d_udp2 { udp("10.0.0.2" port(1999)); };

filter messages from facility local1 and level info to warning
filter f_local1 { facility(local1) and level(info..warn); };

filter messages from facility local 1 and level err to alert
filter f_critic { facility(local1) and level(err .. alert); };

send info, notice and warning messages to remote server udp1
log { source(src); filter(f_local1); destination(d_udp1); };

send error, critical and alert messages to remote server udp2
log { source(src); filter(f_critic); destination(d_udp2); };
```



# Chapter 3 - Additional Features

---

**Step 3: Select the Serial Ports link.**

Click on the Serial Ports link on the Link Panel to the left of the page or in the Configuration section of the page. This will take you to the Port Selection page.

**Step 4: Select port(s).**

On the Port Selection page, choose all ports or an individual port to configure, from the dropdown menu. Click the Submit button. This will take you to the Serial Port Configuration page.

**Step 5: Scroll down to the Terminal Server section.**

You can change the settings for Banner Field (issue) and Login Prompt field here.

## Wizard Method

**Step 1: Bring up the wizard.**

At the command prompt, type the following to bring up the Terminal Appearance custom wizard:

```
wiz --tl
```

Screen 1 will appear.

*Screen 1:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

```
Set to defaults? (y/n) [N]:
```

# Chapter 3 - Additional Features

---

---

## *Screen 2:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

### INSTRUCTIONS:

You can:

- 1) Enter the appropriate information for your system and press ENTER. Enter '#' if you want to deactivate that parameter or
- 2) Press ENTER if you are satisfied with the value within the brackets [ ] and want to go on to the next parameter or
- 3) Press ESC if you want to exit.

ALL.ISSUE - This text determines the format of the login banner that is issued when a connection is made to the system. \n represents a new line and \r represents a carriage return.

```
all.issue:\r\n\
Welcome to terminal server %h port S%p \n\
\r\
\r\n\ Customer Support: 510-770-9727

www.cyclades.com/\n\
\r\n
```

ALL.PROMPT - This text defines the format of the login prompt.

```
all.prompt[%h login:] :
```

# Chapter 3 - Additional Features

---

## *Screen 3:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Your current configuration parameters are:  
(The ones with the '#' means it's not activated.)

```
all.issue : \r\n\
Welcome to terminal server %h port S%p \n\
\r\n\

\r\n\ Customer Support: 510-770-9727

www.cyclades.com/\n\

\r\n\

all.prompt : %h login:
```

Are these configuration(s) all correct (Y)es or (N)o [N] :

### *If you type 'N'*

Type 'c' to go back and CORRECT these parameters  
or 'q' to QUIT :

***Typing 'c' repeats the application, typing 'q' exits the entire wiz application***

### *If you type 'Y'*

Type 'c' to CONTINUE to set these parameters for  
specific ports or 'q' to QUIT :

***Typing 'c' leads to Screen 4, typing 'q' leads to Screen 5.***

## *Screen 4:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

You have 8 available ports on this system.

Type 'q' to quit, a valid port number[1-8], or anything  
else to refresh :

# Chapter 3 - Additional Features

---

---



**NOTE:** The number of available ports depends on the system you are on. Typing in a valid port number repeats this program except this time it's configuring for the port number you have chosen. Typing 'q' leads to Screen 5.

## *Screen 5:*

```

***** C O N F I G U R A T I O N W I Z A R D *****

```

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

If you choose to save to flash, your configurations thus far will still be in the memory of the system even after you reboot it. If you don't save to flash and if you were to reboot the system, all your new configurations will be lost and you will have to reconfigure the system.

Do you want to save your configurations to flash (Y/N) [N]

# Chapter 3 - Additional Features

---

## Time Zone

The content of the file `/etc/TIMEZONE` can be in one of two formats. The first format is used when there is no daylight savings time in the local time zone:

```
std offset
```

The *std* string specifies the name of the time zone and must be three or more alphabetic characters. The offset string immediately follows *std* and specifies the time value to be added to the local time to get *Coordinated Universal Time* (UTC). The offset is positive if the local time zone is west of the Prime Meridian and negative if it is east. The hour must be between 0 and 24, and the minutes and seconds must be between 0 and 59.

The second format is used when there is daylight savings time:

```
std offset dst [offset],start[/time],end[/time]
```

There are no spaces in the specification. The initial *std* and *offset* specify the Standard Time zone, as described above. The *dst* string and *offset* specify the name and offset for the corresponding daylight savings time zone. If the *offset* is omitted, it defaults to one hour ahead of Standard Time.

The *start* field specifies when daylight savings time goes into effect and the *end* field specifies when the change is made back to Standard Time. These fields may have the following formats:

- Jn* This specifies the Julian day, with *n* being between 1 and 365. February 29 is never counted even in leap years.
- n* This specifies the Julian day, with *n* being between 1 and 365. February 29 is counted in leap years.
- Mm.w.d* This specifies day, *d* ( $0 < d < 6$ ) of week *w* ( $1 < w < 5$ ) of month *m* ( $1 < m < 12$ ). Week 1 is the first week in which day *d* occurs and week 5 is the last week in which day *d* occurs. Day 0 is a Sunday.

The time fields specify when, in the local time currently in effect, the change to the other time occurs. If omitted, the default is 02:00:00.

# Chapter 3 - Additional Features

---

---

In the example below:

```
GST+7DST+6M4.1.0/14:30.M10.5.6/10
```

Daylight Savings Time starts on the first Sunday of April at 2:30 p.m. and it ends on the last Saturday of October at 10:00 a.m.

## How to set Date and Time

The date command prints or sets the system date and time. Format of command:

```
date [MMDDhhmm[[CC]YY]
^ ^ ^ ^ ^ ^
^ ^ ^ ^ ^ year
^ ^ ^ ^ century
^ ^ ^ minute
^ ^ hour
^ day
month
```

For example:

```
date 101014452002
```

produces:

```
Thu Oct 10 14:45:00 DST 2002
```

The DST is because it was specified in `/etc/TIMEZONE`



# Chapter 3 - Additional Features

---

This page has been left intentionally blank.

# Appendix A - New User Background Information

---

---

## Users and Passwords

A username and password are necessary to log in to the AlterPath Console Server. The user *root* is predefined, with a password *tslinux*. A password should be configured as soon as possible to avoid unauthorized access. Type the command:

```
passwd
```

to create a password for the root user. To create a regular user (without root privileges), use the commands:

```
adduser user_name
```

```
passwd user_password
```

To log out, type “logout” at the command prompt.

## Linux File Structure

The Linux file system is organized hierarchically, with the base (or root) directory represented by the symbol “/”. All folders and files are nested within each other below this base directory. The directories located just below the base directory are:

- /home*     Contains the work directories of system users.
- /bin*       Contains applications and utilities used during system initialization.
- /dev*       Contains files for devices and ports.
- /etc*       Contains configuration files specific to the operating system.
- /lib*       Contains shared libraries.
- /proc*      Contains process information.
- /mnt*       Contains information about mounted disks.
- /opt*       Location where packages not supplied with the operating system are stored.

# Appendix A - New User Background Information

---

---

- /tmp* Location where temporary files are stored.
- /usr* Contains most of the operating system files.
- /var* Contains operating system data files.

## Basic File Manipulation Commands

The basic file manipulation commands allow the user to copy, delete, and move files and create and delete directories.

- |                                          |                                                                                                                                   |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| <i>cp file_name destination</i>          | Copies the file indicated by <i>file_name</i> to the path indicated by <i>destination</i> .                                       |
| a) <i>cp text.txt /tmp</i>               | a) Copies the file <i>text.txt</i> in the current directory to the <i>tmp</i> directory.                                          |
| b) <i>cp /chap/robo.php ./excess.php</i> | b) Copies the file <i>robo.php</i> in the <i>chap</i> directory to the current directory and renames the copy <i>excess.php</i> . |
| <i>rm file_name</i>                      | Removes the file indicated by <i>file_name</i> .                                                                                  |
| <i>mv file_name destination</i>          | Moves the file indicated by <i>file_name</i> to the path indicated by <i>destination</i> .                                        |
| <i>mkdir directory_name</i>              | Creates a directory named <i>directory_name</i> .                                                                                 |
| a) <i>mkdir spot</i>                     | a) creates the directory <i>spot</i> in the current directory.                                                                    |
| b) <i>mkdir /tmp/snuggles</i>            | b) creates the directory <i>snuggles</i> in the directory <i>tmp</i> .                                                            |
| <i>rmdir directory_name</i>              | Removes the directory indicated by <i>directory_name</i> .                                                                        |

Other commands allow the user to change directories and see the contents of a directory.

- |            |                                                                                                                                                               |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>pwd</i> | Supplies the name of the current directory. While logged in, the user is always “in” a directory. The default initial directory is the user's home directory. |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|

# Appendix A - New User Background Information

---

---

`/home/ <username>` `s [options] directory_name` Lists the files and directories within `directory_name`. Some useful options are `-l` for more detailed output and `-a` which shows hidden system files.

`cd directory_name` Changes the directory to the one specified.

`cat file_name` Prints the contents of `file_name` to the screen.

## Shortcuts:

`.` (one dot) Represents the current directory.

`..` (two dots) Represents one directory above the current directory (i.e. one directory closer to the base directory).

## The vi Editor

To edit a file using the vi editor, type:

```
vi file_name
```

Vi is a three-state line editor: it has a command mode, a line mode and an editing mode. If in doubt as to which mode you are in, press the `<ESC>` key which will bring you to the command mode.

Table 12: vi modes

| Mode         | What is done there               | How to get there                        |
|--------------|----------------------------------|-----------------------------------------|
| Command mode | Navigation within the open file. | Press the <code>&lt;ESC&gt;</code> key. |
| Editing mode | Text editing.                    | See list of editing commands below.     |

# Appendix A - New User Background Information

---

---

Table 12: vi modes

| Mode      | What is done there                             | How to get there                            |
|-----------|------------------------------------------------|---------------------------------------------|
| Line mode | File saving, opening, etc.<br>Exiting from vi. | From the command mode, type ":"<br>(colon). |

When you enter the vi program, you are automatically in command mode. To navigate to the part of the file you wish to edit, use the following keys:

Table 13: vi navigation commands

|          |                                                   |
|----------|---------------------------------------------------|
| <i>h</i> | Moves the cursor to the left (left arrow).        |
| <i>j</i> | Moves the cursor to the next line (down arrow).   |
| <i>k</i> | Moves the cursor to the previous line (up arrow). |
| <i>l</i> | Moves the cursor to the right (right arrow).      |

Having arrived at the location where text should be changed, use these commands to modify the text (note commands "i" and "o" will move you into edit mode and everything typed will be taken literally until you press the <ESC> key to return to the command mode).

Table 14: vi file modification commands

|           |                                                                                                   |
|-----------|---------------------------------------------------------------------------------------------------|
| <i>i</i>  | Inserts text before the cursor position (everything to the right of the cursor is shifted right). |
| <i>o</i>  | Creates a new line below the current line and insert text (all lines are shifted down).           |
| <i>dd</i> | Removes the entire current line.                                                                  |
| <i>x</i>  | Deletes the letter at the cursor position.                                                        |

After you have finished modifying a file, enter line mode (by typing ":" from command mode) and use one of the following commands:

# Appendix A - New User Background Information

---

---

Table 15: vi line mode commands

|               |                                                    |
|---------------|----------------------------------------------------|
| w             | Saves the file (w is for write).                   |
| wq            | Saves and closes the file (q is for quit).         |
| q!            | Closes the file without saving.                    |
| w <i>file</i> | Saves the file with the name <i>&lt;file&gt;</i> . |
| e <i>file</i> | Opens the file named <i>&lt;file&gt;</i> .         |

## The Routing Table

The AlterPath Console Server has a static routing table that can be seen using the commands:

```
route
```

or

```
netstat -rn
```

The file `/etc/network/st_routes` is the AlterPath Console Server's method for configuring static routes. Routes should be added to the file (which is a script run when the AlterPath Console Server is initialized) or at the prompt (for temporary routes) using the following syntax:

```
route [add|del] [-net|-host] target netmask nt_msk [gw gt_way]
interf
```

***[add|del]*** One of these tags must be present. Routes can be either added or deleted.

***[-net|-host]*** Net is for routes to a network and -host is for routes to a single host.

***target*** Target is the IP address of the destination host or network.

# Appendix A - New User Background Information

---

---

- netmask*      The tag *netmask* and *nt\_mask* are necessary only when subnetting is used, otherwise, a mask appropriate to the target is assumed. *nt\_msk* must be specified in dot notation.
- nt\_msk*
- gw gt\_way*    Specifies a gateway, when applicable. *gt\_way* is the IP address or hostname of the gateway.
- interf*        The interface to use for this route. Must be specified if a gateway is not. When a gateway is specified, the operating system determines which interface is to be used.

## Secure Shell Session

Ssh is a command interface and protocol often used by network administrators to connect securely to a remote computer. Ssh replaces its non-secure counterpart rsh and rlogin. There are two versions of the protocol, ssh and ssh2. The AlterPath Console Server offers both. The command to start an ssh client session from a UNIX workstation is:

```
ssh -t <user>@<hostname>
```

where

```
<user> = <username>:ttySnn or
 <username>:socket_port or
 <username>:ip_addr or
 <username>:serverfarm
```

Note: “serverfarm” is a physical port alias. It can be configured in the file *pslave.conf*.

# Appendix A - New User Background Information

---

An example:

```
username: cyclades
ACS16 IP address: 192.168.160.1
host name: acs16
servername for port 1: file_server
```

**ttyS1 is addressed by IP 10.0.0.1 or socket port 7001. The various ways to access the server connected to the port are:**

```
ssh -t cyclades:ttyS1@acs16
ssh -t cyclades:7001@acs16
ssh -t cyclades:10.0.0.1@acs16
ssh -t cyclades:file_server@acs16
ssh -t -l cyclades:10.0.0.1acs16
ssh -t -l cyclades:7001 acs16
```

**For openssh clients, version 3.1p1 or later ssh2 is the default. In that case, the -l flag is used for ssh1.**

```
ssh -t cyclades:7001@acs16
(openssh earlier than 3.1p1 - Cyclades-TS V_1.3.1 and earlier -> ssh1 will be used)
```

```
ssh -t -2 cyclades:7001@acs16
(openssh earlier than 3.1p1 - Cyclades-TS V_1.3.1 and earlier -> ssh2 will be used)
```

```
ssh -t cyclades:7001@acs16
(openssh 3.1p1 or later - Cyclades-TS V_1.3.2 or later/AlterPath Console Server version 2.1.0 or later -> ssh2 will be used)
```



# Appendix A - New User Background Information

---

---

```
ssh -t -l cyclades:7001@acs16
```

(openssh 3.1p1 or later - Cyclades-TS V\_1.3.2 or later/AlterPath Console Server version 2.1.0 or later -> ssh1 will be used)

To log in to a port that does not require authentication, the username is not necessary:

```
ssh -t -2 :ttyS1@acs16
```

**Note:** In this case, the file `sshd_config` must be changed in the following way:

```
PermitRootLogin Yes
```

```
PermitEmptyPassword Yes
```

Configuring `sshd`'s client authentication using SSH Protocol version 1

**Step 1: Only `RhostsAuthentication` yes in `sshd_config`.**

- One of these:

```
hostname or ipaddress in /etc/hosts.equiv or /etc/ssh/
shosts.equiv
```

```
hostname or ipaddress and username in ~/.rhosts or ~/.shosts
and IgnoreRhosts no in sshd_config
```

- Client start-up command: `ssh -t <(A)CTS_ip or Serial_port_ip>` (if the ssh client is running under a session belonging to a username present both in the workstation's database and the (A)CS's database).
- Client start-up command: `ssh -t -l <username> <(A)CTS_ip or Serial_port_ip>` (if the ssh client is running under a session belonging to a username present only in the workstation's database. In this case, the `<username>` indicated would have to be a username present in the (A)CS's database).



**Note:** For security reasons, some ssh clients do not allow just this type of authentication. To access the serial port, the (A)CS must be configured for local authentication. No root user should be used as username.

# Appendix A - New User Background Information

---

Step 2: Only `RhostsRSAAuthentication` yes in `sshd_config`.

- One of the `RhostsAuthentication` settings, described in Step 1.
- Client machine's host key (`SETC/ssh_host_key.pub`) copied into the `TS/tmp/known_hosts` file. The client hostname plus the information inside this file must be appended in one single line inside the file `/etc/ssh/ssh_known_hosts` or `~/.ssh/known_hosts` and `IgnoreUserKnownHosts` no inside `sshd_config`. The following commands can be used for example:

```
echo `n `client_hostname ` >> /etc/ssh/ssh_known_hosts or ~/.ssh/known_hosts
```

```
cat /tmp/known_hosts >> /etc/ssh/ssh_known_hosts or ~/.ssh/known_hosts
```

- client start-up command: `ssh -t <(A)CTS_ip or Serial_port_ip>`



Note: “client\_hostname” should be the DNS name. To access the serial port, the (A)CS must be configured for local authentication. No root user should be used as username.

Step 3: Only `RSAAuthentication` yes in `sshd_config`.

- Removal of the (A)CS's `*.equiv`, `~/.?hosts`, and `*known_hosts` files.
- Client identity created by `ssh-keygen` and its public part (`~/.ssh/identity.pub`) copied into (A)CS's `~/.ssh/authorized_keys`.
- Client start-up command: `ssh -t <(A)CTS_ip or Serial_port_ip>`.

Step 4: Only `PasswdAuthentication` yes in `sshd_config`.

- Removal of the (A)CS's `*.equiv`, `~/.?hosts`, `*known_hosts`, and `*authorized_keys` files.
- Client startup command: `ssh -t -l <username> <(A)CTS_ip or Serial_port_ip> or ssh -t -l <username:alias><(A)CTS_ip>`.

# Appendix A - New User Background Information

Configuring sshd's client authentication using SSH Protocol version 2

Only `PasswdAuthentication yes` in `sshd_config` DSA Authentication is the default. (Make sure the parameter `PubkeyAuthentication` is enabled.)

- Client DSA identity created by `ssh-keygen -d` and its public part (`~/.ssh/id_dsa.pub`) copied into the (A)CS's `~/.ssh/authorized_keys2` file.
- Password Authentication is performed if DSA key is not known to the (A)CS. Client start-up command: `ssh -2 -t <TS_ip or Serial_port_ip>`.



Note: All files “~/\*” or “~/.ssh/\*” must be owned by the user and readable only by others. All files created or updated must have their full path and file name inside the file `config_files` and the command `saveconf` must be executed before rebooting the (A)CS.

## The Process Table

The process table shows which processes are running. Type `ps -a` to see a table similar to that below.

Table 16: Process table

| PID | UID  | State | Command                  |
|-----|------|-------|--------------------------|
| 1   | root | S     | /sbin/inetd              |
| 31  | root | S     | /sbin/sshd               |
| 32  | root | S     | /sbin/cy_ras             |
| 36  | root | S     | /sbin/cy_wdt_led wdt led |
| 154 | root | R     | /ps -a                   |

# Appendix A - New User Background Information

---

To restart the `cy_ras` process use its process ID or execute the command:

```
signal_ras hup
```

This executes the `ps` command, searches for the `cy_ras` process id, then sends the signal `hup` to the process, all in one step. Never kill `cy_ras` with the signals `-9` or `SIGKILL`.

## TS Menu Script

The `ts_menu` script can be used to avoid typing long telnet or ssh commands. It presents a short menu with the names of the servers connected to the serial ports of the AlterPath Console Server. The server is selected by its corresponding number. `ts_menu` must be executed from a local session: via console, telnet, ssh, dumb terminal connected to a serial port, etc. Only ports configured for console access (protocols `socket_server` or `socket_ssh`) will be presented. To start having familiarity with this application, run `ts_menu - h`:

```
> ts_menu -h
```

```
USAGE: ts_menu options
```

```
-p : Display Ethernet Ip and Tcp port
```

```
-i : Display local Ip assigned to the serial port
```

```
-u <name> : Username to be used in ssh/telnet command
```

```
-U : Allows choosing of different usernames for different ports
```

```
-h : print this help message
```

```
> ts_menu
```

```
Master and Slaves Console Server Connection Menu
```

```
1 TSJen800
```

```
2 edson-r4.Cyclades.com
```

```
3 az84.Cyclades.com
```

```
4 64.186.190.85
```

# Appendix A - New User Background Information

---

---

```
5 az85.Cyclades.com
```

```
Type 'q' to quit, a valid option [1-5], or anything else to refresh:
```

By selecting 1 in this example, the user will access the local serial ports on that AlterPath Console Server. If the user selects 2 through 5, remote serial ports will be accessed. This is used when there is clustering (one AlterPath Console Server master box and one or more AlterPath Console Server slave boxes).

If the user selects 1, the following screen is displayed:

```
Serial Console Server Connection Menu for your Master Terminal Server
```

```
1 ttyS1 2 ttyS2 3 s3serverfarm
```

```
Type 'q' to quit, 'b' to return to previous menu, a valid option[1-3], or anything else to refresh:
```

Options 1 to 3 in this case are serial ports configured to work as a CAS profile. Serial port 3 is presented as an alias name (s3serverfarm). When no name is configured in pslave.conf, ttyS<N> is used instead. Once the serial port is selected, the username and password for that port (in case there is a per-user access to the port and -U is passed as parameter) will be presented, and access is granted.

To access remote serial ports, the presentation will follow a similar approach to the one used for local serial ports.

The ts\_menu script has the following line options:

-p : Displays Ethernet IP Address and TCP port instead of server names.

```
AlterPath Console Server: Serial Console Server Connection menu
```

```
1 209.81.55.79 7001 2 209.81.55.79 7002 3 209.81.55.79 7003
```

```
4 209.81.55.79 7004 5 209.81.55.79 7005 6 209.81.55.79 7006
```

```
Type 'q' to quit, a valid option [1-6], or anything else to refresh :
:
```

# Appendix A - New User Background Information

---

**-i** : Displays Local IP assigned to the serial port instead of server names.

AlterPath Console Server: Serial Console Server Connection menu

1 192.168.1.101 2 192.168.1.102 3 192.168.1.103 4 192.168.1.104

5 192.168.1.105 6 192.168.1.106

Type 'q' to quit, a valid option [1-6], or anything else to refresh  
:

**-u <name>** : Username to be used in the ssh/telnet command. The default username is that used to log onto the AlterPath Console Server.

**-h** : Lists script options.

# Appendix B - Cabling, Hardware, & Electrical

## General Hardware Specifications

The power requirements, environmental conditions and physical specifications of the AlterPath Console Server are listed below.

**Table 17: AlterPath Console Server power requirements**

| Power Specifications  |                                                                                          |                                                                                          |                                                         |                                                         |                               |
|-----------------------|------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|---------------------------------------------------------|---------------------------------------------------------|-------------------------------|
|                       | ACS4                                                                                     | ACS8                                                                                     | ACS16                                                   | ACS32                                                   | ACS48                         |
| Input Voltage Range   | External Universal Input Desktop Power Supply (100-240VAC auto-range input, 5VDC output) | External Universal Input Desktop Power Supply (100-240VAC auto-range input, 5VDC output) | Internal 100-240VAC autorange (-48VDC option available) | Internal 100-240VAC autorange (-48VDC option available) | Internal 100-240VAC autorange |
| Input Frequency Range | 50/60H                                                                                   | 50/60H                                                                                   | 50/60H                                                  | 50/60H                                                  | 50/60H                        |
| Power @120VAC         | 5 W max                                                                                  | 6 W max                                                                                  | 22 W max                                                | 26 W max                                                | 11 W max                      |
| Power @220 VAC        | 6 W max                                                                                  | 8 W max                                                                                  | 28 W max                                                | 37 W max                                                | 17 W max                      |

**Table 19: AlterPath Console Server environmental conditions**

| Environmental Information |                            |                            |                            |                            |                            |
|---------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|
|                           | ACS4                       | ACS8                       | ACS16                      | ACS32                      | ACS48                      |
| Operating Temperature     | 50F to 112F (10°C to 50°C) | 50F to 112F (10°C to 50°C) | 50F to 112F (10°C to 50°C) | 50F to 112F (10°C to 50°C) | 50F to 112F (10°C to 50°C) |
| Relative Humidity         | 10 - 90%, non-condensing   | 10 - 90%, non-condensing   | 10 - 90%, non-condensing   | 10 - 90%, non-condensing   | 10 - 90%, non-condensing   |

# Appendix B - Cabling, Hardware, & Electrical

Table 20: AlterPath CS physical specifications

| Physical Information |                            |                            |                             |                             |                             |
|----------------------|----------------------------|----------------------------|-----------------------------|-----------------------------|-----------------------------|
|                      | ACS4                       | ACS8                       | ACS16                       | ACS32                       | ACS48                       |
| External Dimensions  | 8.5 in. x 4.75 in. x 1 in. | 8.5 in. x 4.75 in. x 1 in. | 17 in. x 8.5 in. x 1.75 in. | 17 in. x 8.5 in. x 1.75 in. | 17 in. x 8.5 in. x 1.75 in. |
| Weight               | 1.5 lb.                    | 1.6 lb.                    | 6 lb.                       | 6.2 lb.                     | 8 lb.                       |

Table 21: AlterPath Console Server safety specifications

| Safety Information |                     |      |       |       |       |
|--------------------|---------------------|------|-------|-------|-------|
|                    | ACS4                | ACS8 | ACS16 | ACS32 | ACS48 |
| Approvals          | FCC and CE, Class A |      |       |       |       |

This section has all the information you need to quickly and successfully purchase or build cables to the AlterPath Console Server. It focuses on information related to the RS-232 interface, which applies not only to the AlterPath Console Server but also to any RS-232 cabling.

## The RS-232 Standard

RS-232C, EIA RS-232, or simply RS-232 refer to a standard defined by the Electronic Industries Association in 1969 for serial communication. More than 30 years later, more applications have been found for this standard than its creators could have imagined. Almost all electronic devices nowadays have serial communication ports.

RS-232 was defined to connect Data Terminal Equipment, (DTE, usually a computer or terminal) to Data Communication Equipment (DCE, usually a modem):



# Appendix B - Cabling, Hardware, & Electrical

---

DTE > RS-232 > DCE > communication line > DCE > RS-232 > DTE

RS-232 is now mostly being used to connect DTE devices directly (without modems or communication lines in between). While that was not the original intention, it is possible with some wiring tricks. The relevant signals (or wires) in a RS-232 cable, from the standpoint of the computer (DTE), are:

|                                                   |                                                                     |
|---------------------------------------------------|---------------------------------------------------------------------|
| <i>Receive Data (RxD) and Transmit Data (TxD)</i> | The actual data signals                                             |
| <i>Signal Ground (Gnd)</i>                        | Electrical reference for both ends                                  |
| <i>Data Terminal Ready (DTR)</i>                  | Indicates that the computer (DTE) is active                         |
| <i>Data Set Ready (DSR)</i>                       | Indicates that the modem (DCE) is active.                           |
| <i>Data Carrier Ready (DCD)</i>                   | Indicates that the connection over the communication line is active |
| <i>CTS (Clear to Send, an input)</i>              | Flow control for data flowing from DTE to DCE                       |
| <i>RTS (Request to Send, an output)</i>           | Flow control for data flowing from DCE to DTE                       |

Not all signals are necessary for every application, so the RS-232 cable may not need all 7 wires. The RS-232 interface defines communication parameters such as parity, number of bits per character, number of stop-bits and the baud rate. Both sides must be configured with the same parameters. That is the first thing to verify if you think you have the correct cable and things still do not work. The most common configuration is 8N1 (8 bits of data per character, no parity bit included with the data, 1 stop-bit to indicate the end of a character). The baud rate in a RS-232 line translates directly into the data speed in bits per second (bps). Usual transmission speeds range between 9,600 bps and 19,200bps (used in most automation and console applications) to 115,200 bps (used by the fastest modems).

# Appendix B - Cabling, Hardware, & Electrical

---

---

## Cable Length

The original RS-232 specifications were defined to work at a maximum speed of 19,200 bps over distances up to 15 meters (or about 50 feet). That was 30 years ago. Today, RS-232 interfaces can drive signals faster and through longer cables.

As a general rule, consider:

- If the speed is lower than 38.4 kbps, you are safe with any cable up to 30 meters (100 feet)
- If the speed is 38.4 kbps or higher, cables should be shorter than 10 meters (30 feet)
- If your application is outside the above limits (high speed, long distances), you will need better quality (low impedance, low-capacitance) cables.

Successful RS-232 data transmission depends on many variables that are specific to each environment. The general rules above are empirical and have a lot of safety margins built-in.

## Connectors

The connector traditionally used with RS-232 is the 25-pin D-shaped connector (DB-25). Most analog modems and most older computers and serial equipment use this connector. The RS-232 interface on DB-25 connector always uses the same standard pin assignment.

The 9-pin D-shaped connector (DB-9) saves some space and is also used for RS-232. Most new PC COM ports and serial equipment (specially when compact size is important) uses this connector. RS-232 interfaces on DB-9 connectors always use the same standard pin assignment.

The telephone-type modular RJ-45 plug and jack are very compact, inexpensive and compatible with the phone and Ethernet wiring systems present in most buildings and data centers. Most networking equipment and new servers use RJ-45 connectors for serial communication. Unfortunately there is no standard RS-232 pin assignment for RJ-45 connectors. Every equipment vendor has its pin assignment.

# Appendix B - Cabling, Hardware, & Electrical

Most connectors have two versions. The ones with pins are said to be “male” and the ones with holes are said to be “female”.

Table 23: Cables and their pin specifications

| RS-232 Signal | Name/Function (Input/Output) | DB-25 pins (Standard) | DB-9 pins (Standard) | RJ-45 pins (Cyclades) |
|---------------|------------------------------|-----------------------|----------------------|-----------------------|
| Chassis       | Safety Ground                | 1                     | Shell                | Shell                 |
| TxD           | Transmit Data (O)            | 2                     | 3                    | 3                     |
| RxD           | Receive Data (I)             | 3                     | 2                    | 6                     |
| DTR           | Data Terminal Ready (O)      | 20                    | 4                    | 2                     |
| DSR           | Data Set Ready (I)           | 6                     | 6                    | 8                     |
| DCD           | Data Carrier Detect (I)      | 8                     | 1                    | 7                     |
| RTS           | Request To Send (O)          | 4                     | 7                    | 1                     |
| CTS           | Clear To Send (I)            | 5                     | 8                    | 5                     |
| Gnd           | Signal Ground                | 7                     | 5                    | 4                     |

## Straight-Through vs. Crossover Cables

The RS-232 interface was originally intended to connect a DTE (computer, printer and other serial devices) to a DCE (modem) using a straight-through cable (all signals on one side connecting to the corresponding signals on the other side one-to-one). By using some “cabling tricks”, we can use RS-232 to connect two DTEs as is the case in most modern applications.

A crossover (a.k.a. null-modem) cable is used to connect two DTEs directly, without modems or communication lines in between. The data signals between the two sides are transmitted and received and there are many variations on how the other control signals are wired. A “complete” crossover cable would connect TxD with RxD, DTR with DCD/DSR, and RTS with CTS on both sides. A “simplified” crossover cable would cross TxD and RxD and locally short-circuit DTR with DCD/DSR and RTS with CTS.

# Appendix B - Cabling, Hardware, & Electrical

---

---

## Which cable should be used?

First, look up the proper cable for your application in the table below. Next, purchase standard off-the-shelf cables from a computer store or cable vendor. For custom cables, refer to the cable diagrams to build your own cables or order them from Cyclades or a cable vendor.

Table 24: Which cable to use

| To Connect To                                                                                                            | Use Cable                                                                                                                                                                                                             |
|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DCE DB-25 Female (standard) <ul style="list-style-type: none"><li>Analog Modems</li><li>ISDN Terminal Adapters</li></ul> | Cable 1:<br>RJ-45 to DB-25 M straight-through (Custom). This custom cable can be ordered from Cyclades or other cable vendors. A sample is included with the product ("straight-through").                            |
| DTE RJ-45 Cyclades (custom) <ul style="list-style-type: none"><li>All Cyclades Console Ports</li></ul>                   | Cable 2:<br>RJ-45 to RJ-45 crossover (custom). A sample is included with the product ("straight-through")<br>This custom cable can be ordered from Cyclades or other cable vendors using the provided wiring diagram. |

# Appendix B - Cabling, Hardware, & Electrical

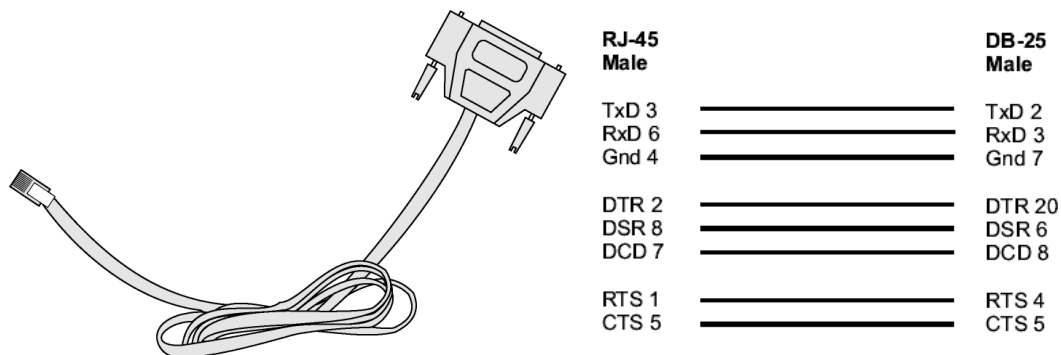
## Cable Diagrams

Before using the following cable diagrams refer to the tables above to select the correct cable for your application. Sometimes, crossover cables are wired slightly differently depending on the application. A “complete” crossover cable would connect the TxD with RxD, DTR with DCD/DSR, and RTS with CTS across both sides. A “simplified” crossover cable would cross TxD and RxD and locally short-circuit DTR with DCD/DSR and RTS with CTS.

Most of the diagrams in this document show the “complete” version of the crossover cables, with support for modem control signals and hardware flow control. Applications that do not require such features have just to configure NO hardware flow control and NO DCD detection on their side. Both ends should have the same configuration for better use of the complete version of the cables.

### Cable #1: Cyclades RJ-45 to DB-25 Male, Straight Through

**Application:** This cable connects Cyclades products (serial ports) to modems and other DCE RS-232 devices. After connecting the appropriate adaptor to the RJ-45 straight-through cable, you will essentially have the cable shown in this picture.

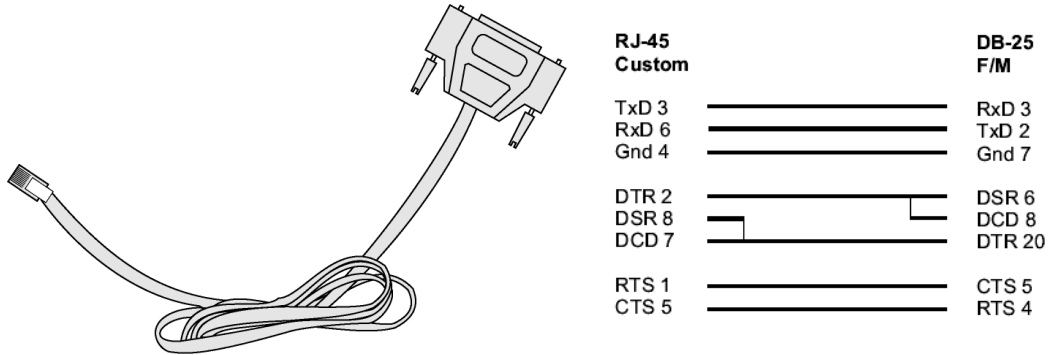


*Figure 32: Cable 1 - Cyclades RJ-45 to DB-25 Male, Straight Through*

# Appendix B - Cabling, Hardware, & Electrical

## Cable #2: Cyclades RJ-45 to DB-25 Female/Male, Crossover

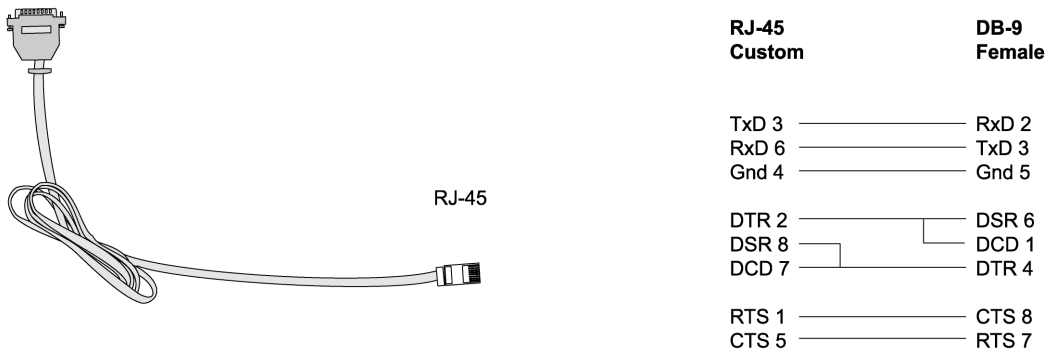
This cable connects Cyclades products (serial ports) to console ports, terminals, printers and other DTE RS-232 devices. After connecting the appropriate adaptor to the RJ-45 straight-through cable, you will essentially have the cable shown in this picture.



*Figure 33: Cable 2 - Cyclades RJ-45 to DB-25 Female/Male, Crossover*

## Cable #3: Cyclades RJ-45 to DB-9 Female, Crossover

This cable connects Cyclades products (serial ports) to console ports, terminals, printers and other DTE RS-232 devices. After connecting the appropriate adaptor to the RJ-45 straight-through cable, you will essentially have the cable shown in this picture.

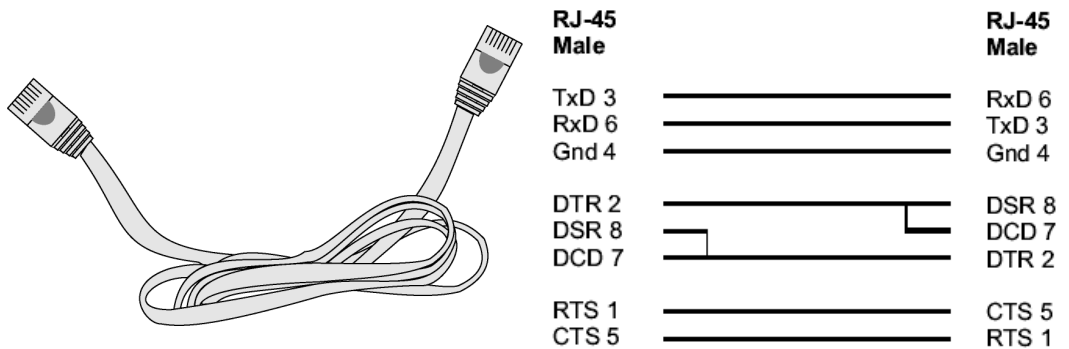


*Figure 34: Cable 3 - Cyclades RJ-45 to DB-9 Female, Crossover*

# Appendix B - Cabling, Hardware, & Electrical

## Cable #4: Cyclades RJ-45 to Cyclades RJ-45, straight-through

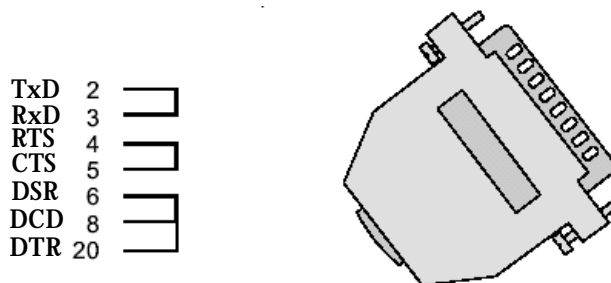
This cable is the main cable that you will use. After configuration it can be used with the same adapter to connect to the server. It can also be used to connect two ports of a Cyclades product ("loopback") for testing purposes?



*Figure 35: Cable 4 - Cyclades RJ-45 to Cyclades RJ-45, Crossover*

## Loop-Back Connector for Hardware Test

The use of the following DB-25 connector is explained in the Troubleshooting chapter.



*Figure 36: Loop-Back Connector*

# Appendix B - Cabling, Hardware, & Electrical

## CAT.5e Inline Coupler/Sun Netra Adapter

This Adapter attaches to the Cyclades RJ-45 to Cyclades RJ-45, Crossover cable. It is usually used in console management applications to connect Cyclades products to a Sun Netra server or to a Cisco product. At one end of the adapter is the black CAT.5e Inline Coupler box with a female RJ-45 terminus, from which a 3-inch-long black “Sun Netra”-labeled cord extends, terminating in an RJ-45 male connector.

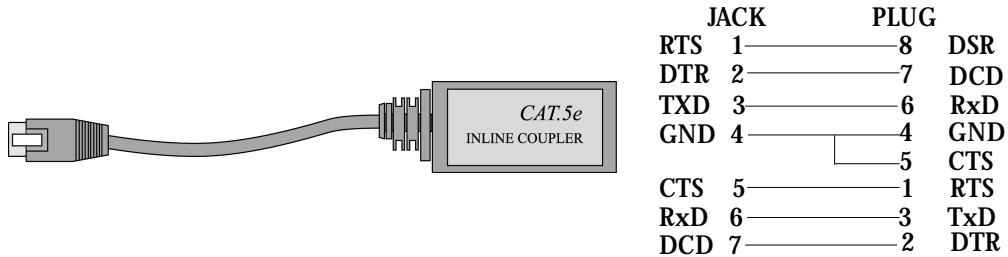


Figure 37: CAT.5e Inline Coupler/Sun Netra Adapter

## Adapters

The following four adapters are included in the product box. A general diagram is provided below and then a detailed description is included for each adapter.

### RJ-45 Female to DB-25 Male Adapter

The following adapter may be necessary.

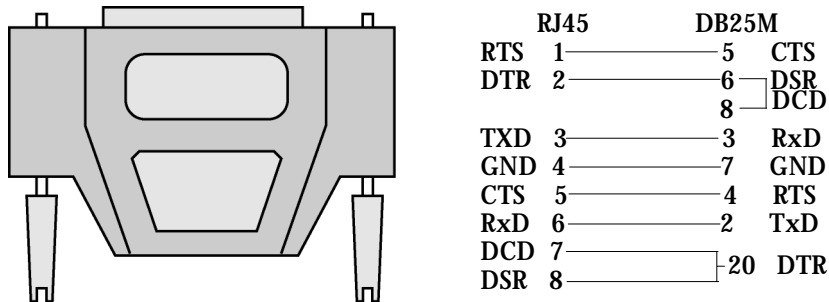


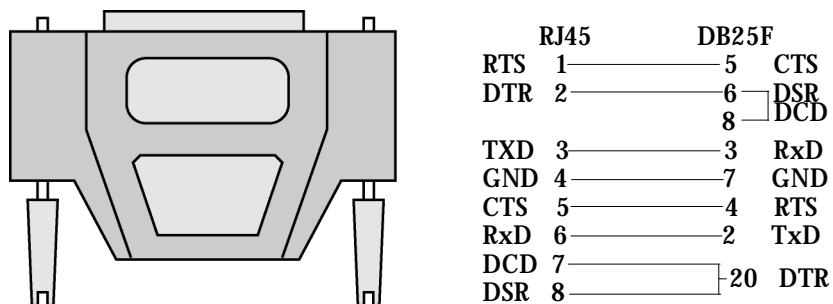
Figure 38: RJ-45 Female to DB-25 Male Adapter



# Appendix B - Cabling, Hardware, & Electrical

## RJ-45 Female to DB-25 Female Adapter

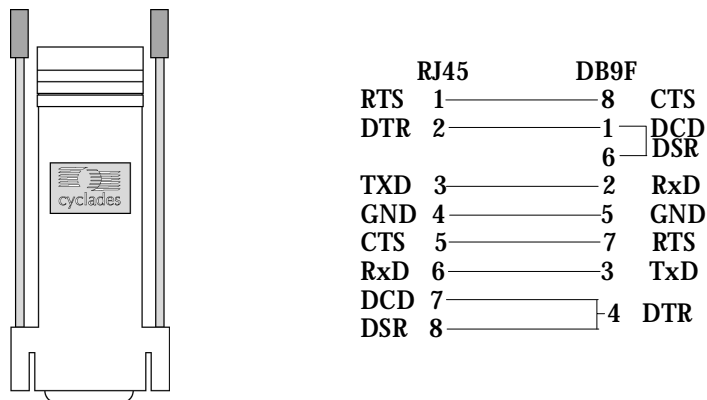
The following adapter may be necessary.



*Figure 39: RJ-45 Female to DB-25 Female Adapter*

## RJ-45 Female to DB-9 Male Adapter

The following adapter may be necessary.



*Figure 40: RJ-45 Female to DB-9 Male Adapter*

# Appendix B - Cabling, Hardware, & Electrical

## RJ-45 Female to DB-9 Female Adapter

The following adapter may be necessary.

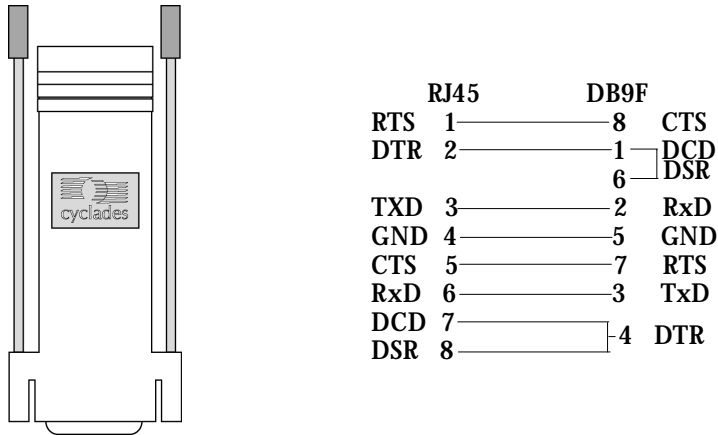


Figure 41: RJ-45 Female to DB-9 Female Adapter

## DB-25 Male to DB-9 Female Adapter

The following adapter may be necessary.

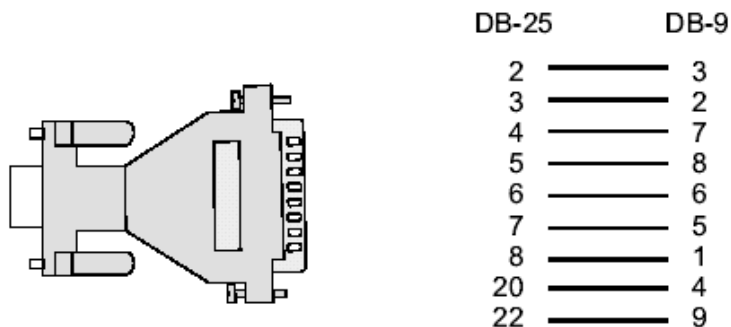


Figure 42: DB-25 Male to DB-9 Female Adapter

# Appendix C - The pslave Configuration File

## Introduction

This chapter begins with the complete table for all parameters and their descriptions. The `pslave.conf` file with all possible parameters and their descriptions follows. You can find samples of the pslave configuration files (`pslave.conf`, `.cas`, `.ts`, and `.ras`) in the `/etc/portslave` directory in the (A)CS box.

## Configuration Parameters

### Additional AlterPath Console Server Options for a CAS

You can configure additional features with the parameters given on the following tables:

Table 27: Parameters Common to CAS, TS, & Dial-in Access

| Parameter                  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Value for this Example |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| <code>conf.eth_ip</code>   | Configured in <a href="#">Task 4: Edit the pslave.conf file in Chapter 2 - Installation and Configuration</a> . This is the IP address of the Ethernet interface. This parameter, along with the next two, is used by the <code>cy_ras</code> program to OVERWRITE the file <code>/etc/network/ifcfg_eth0</code> as soon as the command “ <code>signal_ras hup</code> ” is executed. The file <code>/etc/network/ifcfg_eth0</code> should not be edited by the user unless the <code>cy_ras</code> configuration is not going to be used. | 200.200.200.1          |
| <code>conf.eth_mask</code> | The mask for the Ethernet network.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 255.255.255.0          |
| <code>conf.eth_mtu</code>  | The Maximum Transmission Unit size, which determines whether or not packets should be broken up.                                                                                                                                                                                                                                                                                                                                                                                                                                          | 1500                   |

# Appendix C - The pslave Configuration File

Table 27: Parameters Common to CAS, TS, & Dial-in Access

| Parameter     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Value for this Example |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| conf.lockdir  | The lock directory, which is /var/lock for the AlterPath Console Server. It should not be changed unless the user decides to customize the operating system.                                                                                                                                                                                                                                                                                                   | /var/lock              |
| all.speed     | The speed for all ports.                                                                                                                                                                                                                                                                                                                                                                                                                                       | 9600                   |
| all.datasize  | The data size for all ports.                                                                                                                                                                                                                                                                                                                                                                                                                                   | 8                      |
| all.stopbits  | The number of stop bits for all ports.                                                                                                                                                                                                                                                                                                                                                                                                                         | 1                      |
| all.parity    | The parity for all ports.                                                                                                                                                                                                                                                                                                                                                                                                                                      | none                   |
| all.DTR_reset | Valid only for the CAS configuration. This value specifies how long (in milliseconds) a DTR signal will be turned off before it is turned back on again. If set to 0, this parameter will NOT be active. This may be dangerous if a user were to connect to a port that a previous user was on but had lost the session after a timeout. The user may directly connect into the previous user's shell. A minimum of 100ms is required otherwise it is assumed. | 100                    |
| all.authtype  | Configured in <a href="#">Task 4: Edit the pslave.conf file in Chapter 2 - Installation and Configuration</a> .                                                                                                                                                                                                                                                                                                                                                | radius                 |
| all.authhost1 | This address indicates the location of the Radius/TacacsPlus authentication server and is only necessary if this option is chosen in the previous parameter. A second Radius/TacacsPlus authentication server can be configured with the parameter all.authhost2.                                                                                                                                                                                              | 200.200.200.2          |

# Appendix C - The pslave Configuration File

Table 27: Parameters Common to CAS, TS, & Dial-in Access

| Parameter      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Value for this Example |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.accthost1  | This address indicates the location of the Radius/TacacsPlus accounting server, which can be used to track how long users are connected after being authorized by the authentication server. Its use is optional. If this parameter is not used, accounting will not be performed. If the same server is used for authentication and accounting, both parameters must be filled with the same address. A second Radius/TacacsPlus accounting server can be configured with the parameter all.accthost2. | 200.200.200.2          |
| all.radtimeout | This is the timeout (in seconds) for a Radius/TacacsPlus authentication query to be answered. The first server (authhost1) is tried “radretries” times, and then the second (authhost2), if configured, is contacted “radretries” times. If the second also fails to respond, Radius/TacacsPlus authentication fails.                                                                                                                                                                                   | 3                      |
| all.radretries | Defines the number of times each Radius/TacacsPlus server is tried before another is contacted. The default, if not configured, is 5.                                                                                                                                                                                                                                                                                                                                                                   | 5                      |
| all.secret     | This is the shared secret necessary for communication between the AlterPath Console Server and the Radius/TacacsPlus servers.                                                                                                                                                                                                                                                                                                                                                                           | rad-secret             |
| all.flow       | This sets the flow control to hardware, software, or none.                                                                                                                                                                                                                                                                                                                                                                                                                                              | hard                   |
| all.protocol   | The default CAS setup was explained in Chapter 2, <a href="#">Task 4: Edit the pslave.conf file</a> . The TS configuration settings are in <a href="#">Table 29, “TS Parameters.” on page 251</a> . The Dial-in configuration settings are in <a href="#">Table 30, “Dial-in configuration Parameters.” on page 252</a> .                                                                                                                                                                               | socket_server          |

# Appendix C - The pslave Configuration File

---

---

Table 27: Parameters Common to CAS, TS, & Dial-in Access

| Parameter          | Description                                                                                                                                  | Value for this Example |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.break_sequence | This parameter is the string that is used to send a break to the TTY. It is only valid if TTY protocol is socket_ssh.                        | ~break                 |
| s1.tty             | The device name for the port is set to the value given in this parameter. If a device name is not provided for a port, it will not function. | ttyS1                  |

In addition to the above parameters which are common to all local and remote access scenarios, you can also configure the following parameters for additional options. Many of the parameters are unique to CAS, but some also apply to TS and Dial-in port profiles. This is indicated in these instances.

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                              | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Value for this Example      |
|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|
| conf.nfs_data_buffering                | Remote Network File System where data captured from the serial port will be written instead of the default directory “/var/run/DB”. The directory tree to which the file will be written must be NFS-mounted. If data buffering is turned on for port 1, for example, the data will be stored in the file ttyS1.data (or <serverfarm1>.data if s1.serverfarm was configured) in the directory indicated by this variable (please see also Data Buffering section for more details). The remote host must have NFS installed and the administrator must create, export and allow reading/writing to this directory. The size of this file is not limited by the value of the parameter s1.data_buffering, though the value cannot be zero since a zero value turns off data buffering. The size of the file is dependent on the NFS server only (hard drive, partition size, etc.). | commented                   |
| conf.facility (CAS and Dial-in Access) | This value (0-7) is the Local facility sent to the syslog. The file /etc/syslogng/syslog-ng.conf contains a mapping between the facility number and the action (see more in <a href="#">Generating Alarms</a> in <a href="#">Chapter 3 - Additional Features</a> ).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 7                           |
| conf.DB_facility                       | This value (0-7) is the Local facility sent to the syslog with the data when syslog_buffering is active. The file /etc/syslog-ng/syslog-ng.conf contains a mapping between the facility number and the action (see more on <a href="#">Syslog</a> in Chapter 3).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 0                           |
| conf.group                             | Used to group users to simplify configuration of the parameter all.userslater on. This parameter can be used to define more than one group.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | group_name:<br>user1, user2 |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                                   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Value for this Example |
|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.dcd<br>(CAS and TS)                     | DCD signal (sets the tty parameter CLOCAL). Valid values are 0 or 1. In a socket session, if all.dcd=0, a connection request (telnet or ssh) will be accepted regardless of the DCD signal and the connection will not be closed if the DCD signal is set to DOWN. In a socket connection, if all.dcd=1 a connection request will be accepted only if the DCD signal is UP and the connection (telnet or ssh) will be closed if the DCD signal is set to DOWN. | 0                      |
| all.issue<br>(CAS and TS)                   | This text determines the format of the login banner that is issued when a connection is made to the AlterPath Console Server. \n represents a new line and \r represents a carriage return. Expansion characters can be used here.<br><i>Value for this Example:</i><br>\r\n\ TSLINUX - Portslave Internet Services\n\<br>\r\n\ Welcome to terminal server %h port S%p \n\<br>\r\n\ Customer Support: 510-770-9727<br>www.cyclades.com/\n\<br>\r\n             | See Description column |
| all.prompt<br>(CAS and TS)                  | This text defines the format of the login prompt. Expansion characters can be used here.                                                                                                                                                                                                                                                                                                                                                                       | %h login:              |
| all.ipno<br>(CAS and Dial-in configuration) | This is the default IP address of the AlterPath Console Server's serial ports. The "+" indicates that the first port should be addressed as 192.168.1.101 and the following ports should have consecutive values. Any host can access a port using its IP address as long as a path to the address exists in the host's routing table.                                                                                                                         | 192.168.1.10<br>1+     |



# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Value for this Example |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.poll_interval            | Valid only for protocols socket_server and raw_data. When not set to zero, this parameter sets the wait for a TCP connection keep-alive timer. If no traffic passes through the AlterPath Console Server for this period of time, the AlterPath Console Server will send a line status message to the remote device to see if the connection is still up. If not configured, 1000 ms is assumed (the unit for this parameter is ms). If set to zero, line status messages will not be sent to the socket client.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 0                      |
| all.socket_port (CAS and TS) | <p>In the CAS profile, this defines an alternative labeling system for the AlterPath Console Server ports. The “+” after the numerical value causes the serial interfaces to be numbered consecutively. In this example, serial interface 1 is assigned the port value 7001, serial interface 2 is assigned the port value 7002, etc. One example on how this could be used is in the case of all.protocol or s&lt;n&gt;.protocol socket_ssh and the port value (7001, 7002, etc), if supplied by the ssh client like username:port value, the ssh client will be directly connected with the serial interface.</p> <p>For TS, the all.socket_port (s&lt;n&gt;.socket_port) for the TS profile can be 23 (default value). This means that the TS will initiate a telnet session against a given host. If it is a different value, there will be pure, raw data between the client (TS for that serial port) and the host. The all.protocol (s&lt;n&gt;.protocol HAS to be configured as socket_client. In summary, TS profile (all.protocol is socket_client) raw mode (all.socket_port is NOT 23)</p> | 7001+                  |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Value for this Example |
|--------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.data_buffering (CAS) | A non zero value activates data buffering (local or remote, according to what was configured in the parameter conf.nfs_data_buffering see <a href="#">Data Buffering</a> in Chapter 3). If local data buffering, a file is created on the AlterPath Console Server; if remote, a file is created through NFS in a remote server. All data received from the port is captured in this file. If local data buffering, this parameter means the maximum file size (in bytes). If remote, this parameter is just a flag to activate (greater than zero) or deactivate data buffering. When local data buffering is used, each time the maximum is reached the oldest 10% of stored data is discarded, releasing space for new data (FIFO system) - circular file. When remote data buffering is used, there's no maximum file size other than the one imposed by the remote server - linear file. This file can be viewed using the normal Unix tools (cat, vi, more, etc.). <i>Size is in bytes not kilobytes.</i> See <a href="#">Data Buffering</a> for details. | 0                      |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter              | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Value for this Example |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.DB_mode            | When configured as cir for circular format, the buffer works like a revolving file at all times. The file is overwritten whenever the limit of the buffer size (as configured in all.data_buffering or s<n>.data_buffering) is reached. As for linear format (lin), once the limit of the kernel buffer size is reached (4k), a flow control stop (RTS off or XOFF-depending on how all.f low or s<n>.flow is set) is issued automatically to the remote device so that it will stop sending data to the serial port. Then, when a session is established to the serial port, the data in the buffer is shown to the user if not empty (dont_show_DBmenu parameter assumed to be 2), cleared, and a flow control start (RTS on or XON) is issued to resume data transmission. Once exiting the session, linear data buffering resumes. If all.flow or s<n>.flow is set to none, linear buffering is not possible as there is no way to stop reception through the serial line. Default is cir." | cir                    |
| all.DB_timestamp (CAS) | Records the time stamp in the data buffering file (1) or not (0). If it is configured as 1, the software will accumulate input characters until it receives a CR an LF from the serial port or the accumulated data reaches 256 characters. Either way, the accumulated data will be recorded in the data buffering file along with the current time. The parameter all.data_buffering has to be with a non-zero value for this parameter to be meaningful.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 0                      |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                   | Value for this Example |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.syslog_buffering (CAS) | When non zero, the contents of the data buffer are sent to the syslogng every time a quantity of data equal to this parameter is collected. The syslog level for data buffering is hard coded to level 5 (notice) and facility conf.DB_facility. The file /etc/syslog-ng/syslog-ng.conf should be set accordingly for the syslog-ng to take some action. (See <a href="#">Syslog-ng Configuration to use with Syslog Buffering Feature.</a> ) | 0                      |
| all.dont_show_DBmenu (CAS) | When zero, a menu with data buffering options is shown when a nonempty data buffering file is found. When 1, the data buffering menu is not shown. When 2, the data buffering menu is not shown but the data buffering file is shown if not empty. When 3, the data buffering menu is shown, but without the erase and show and erase options.                                                                                                | 1                      |
| all.alarm (CAS)            | When non zero, all data received from the port are captured and sent to syslog-ng with DAEMON facility and ALERT level. The syslogng.conf file should be set accordingly, for the syslog-ng to take some action (please see <a href="#">Generating Alarms</a> in Chapter 3 - Additional Features for the syslog-ng configuration file).                                                                                                       | 0                      |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Value for this Example  |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| all.users (CAS)      | Restricts access to ports by user name (only the users listed can access the port or, using the character “!”, all but the users listed can access the port.) In this example, the users joe, mark and members of user_group cannot access the port. A single comma and spaces/tabs may be used between names. A comma may not appear between the “!” and the first user name. The users may be local, Radius or TacacsPlus. User groups (defined with the parameter conf.group) can be used in combination with user names in the parameter list. Notice that these are common users, not administrators. | ! joe, mark, user_group |
| all.sniff_mode (CAS) | This parameter determines what other users connected to the very same port (see parameter admin_users below) can see of the session of the first connected user (main session): <i>in</i> shows data written to the port, <i>out</i> shows data received from the port, and <i>i/o</i> shows both streams. The second and later sessions are called sniff sessions and this feature is activated whenever the protocol parameter is set to socket_ssh or socket_server.                                                                                                                                    | out                     |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter                   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Value for this Example  |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| all.admin_users (CAS)       | This parameter determines which users can open a sniff session, which is where other users connected to the very same port can see everything that a first user connected is doing. The other users connected to the very same port can also cancel the first user's session (and take over). If all.multiple_sessions (seen below) is configured as no only two users can connect to the same port simultaneously. If all.multiple_sessions is configured as yes more simultaneous users can sniff the session or have read and/or write permission (please see details in <a href="#">Session Sniffing</a> in Chapter 3). When users want access per port to be controlled by administrators, this parameter is obligatory and authtype must not be none. This parameter can determine who can open a sniff session or cancel a previous session. User groups (defined with the parameter conf.group) can be used in combination with user names in the parameter list. | peter, john, user_group |
| all.multiple_sessions (CAS) | Valid for all serial ports; must be "yes" or "no". If it is not defined, the default will be "no". Please see <a href="#">Session Sniffing</a> in Chapter 3 for details.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | no                      |
| all.escape_char (CAS)       | This parameter determines which character must be typed to make the session enter "menu mode". The possible values are <CTRL-a> to <CTRL-z>. Represent the CTRL with '^'. This parameter is only valid when the port protocol is socket_server or socket_ssh. Default value is '^z'.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | ^z                      |
| all.tx_interval (CAS)       | Valid for protocols socket_server and raw_data. Defines the delay (in milliseconds) before transmission to the Ethernet of data received through a serial port. If not configured, 100ms is assumed. If set to zero or a value above 1000, no buffering will take place.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 100                     |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Value for this Example |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.idletimeout (CAS) | Valid only for the CAS configuration (protocols socket_server, socket_ssh, and raw_data). Specifies how long (in minutes) a connection can remain inactive before it is cut off. If set to zero (the default), the connection will not time out.                                                                                                                                                                                                                                                                                                                                                | 0                      |
| all.sttyCmd (CAS)     | Tty settings after a socket connection to that serial port is established. The tty is programmed to work as a CAS configuration and this user specific configuration is applied over that serial port. Parameters must be separated by space. (e.g., the following example sets igncr which tells the terminal not to ignore the carriage-return on input, -onlcr do not map newline character to a carriage return/newline character sequence on output, opost post-process output, -icrnl do not map carriage-return to a newline character on input. all.sttyCmd -igncr -onlcr opost -icrnl) | commented              |

# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Value for this Example |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.authtype        | <p>Type of authentication used. There are several authentication type options:</p> <ul style="list-style-type: none"><li>• <i>local</i> (authentication is performed using the <code>/etc/passwd</code> file)</li><li>• <i>radius</i> (authentication is performed using a Radius authentication server)</li><li>• <i>TacacsPlus</i> (authentication is performed using a TacacsPlus authentication server)</li><li>• <i>none</i></li><li>• <i>local/radius</i> (authentication is performed locally first, switching to Radius if unsuccessful)</li><li>• <i>radius/local</i> (the opposite of the previous option),</li><li>• <i>RadiusDownLocal</i> (local authentication is tried only when the Radius server is down)</li><li>• <i>local/TacacsPlus</i> (authentication is performed locally first, switching to TacacsPlus if unsuccessful)</li><li>• <i>ldap</i> (authentication is performed against an ldap database using an ldap server. The IP address and other details of the ldap server are defined in the file <code>/etc/ldap.conf</code>)</li><li>• <i>kerberos</i> (authentication is performed using a kerberos server. The IP address and other details of the kerberos server are defined in the file <code>/etc/krb5.conf</code>)</li></ul> | local                  |
| s1.serverfarm (CAS) | Alias name given to the server connected to the serial port. <code>Server_connected</code> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | serial1                |



# Appendix C - The pslave Configuration File

Table 28: Mostly CAS-specific Parameters

| Parameter    | Description                                                                                                                                                                                       | Value for this Example |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
|              | Note that this parameter controls the authentication required by the AlterPath Console Server. The authentication required by the device to which the user is connecting is controlled separately |                        |
| s2.tty (CAS) | See the s1.tty entry in the following table.                                                                                                                                                      | ttyS2                  |
| s8.tty (CAS) | See the s1.tty entry in the following table.                                                                                                                                                      | ttyS8                  |

## CAS Setup Scenario

As shown in [Figure 1: Console Access Server diagram](#), our “CAS with local authentication” scenario has either telnet or ssh (a secure shell session) being used.

After configuring desired additional parameters, execute the command `signal_ras hup` to activate the changes. At this point, the configuration should be tested. A step-by-step check list follows:

### Step 1: Create a new user.

Run the `adduser <username>` to create a new user in the local database. Create a password for this user by running `passwd <username>`.

### Step 2: Confirm physical connection.

Make sure that the physical connection between the AlterPath Console Server and the servers is correct. A cross cable (not the modem cable provided with the product) should be used. Please see [Appendix B - Cabling, Hardware, and Electrical Specifications](#) for pin-out diagrams.

### Step 3: Confirm that server is set to same parameters as the (A)CS.

The AlterPath Console Server has been set for communication at 9600 bps, 8N1. The server must also be configured to communicate on the serial console port with the same parameters.

# Appendix C - The pslave Configuration File

---

## Step 4: Confirm routing.

Also make sure that the computer is configured to route console data to its serial console port (Console Redirection).

## Step 5: Telnet the server connected to port 1.

From a server on the LAN (not from the console), try to telnet to the server connected to the first port of the AlterPath Console Server using the following command:

```
telnet 200.200.200.1 7001
```

For both telnet and ssh sessions, the servers can be reached by either:

1. Ethernet IP of the AlterPath Console Server and assigned socket port.

or

2. Individual IP assigned to each port.

If everything is configured correctly, a telnet session should open on the server connected to port 1. If not, check the configuration, follow the steps above again, and check the troubleshooting appendix.

## Step 6: Activate the changes.

Now continue on to [Task 5: Activate the changes](#) through [Task 8: Reboot the AlterPath Console Server](#) listed in [Chapter 2 - Installation and Configuration](#).



**Note:** It is possible to access the serial ports from Microsoft stations using some off-the-shelf packages. Although Cyclades is not liable for those packages, successful tests were done using at least one of them. From the application's viewpoint running on a Microsoft station, the remote serial port works like a regular COM port. All the I/O with the serial device attached to the AlterPath Console Server is done through socket connections opened by these packages and a COM port is emulated to the application.

# Appendix C - The pslave Configuration File

## TS Parameters

The following parameters are unique to a TS setup except where indicated.

Table 29: TS Parameters

| Parameter               | Description                                                                                                                                                                                                                                                                                                            | Value for this Example |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| conf.telnet             | Location of the telnet utility                                                                                                                                                                                                                                                                                         | /bin/telnet            |
| conf.ssh                | Location of the ssh utility.                                                                                                                                                                                                                                                                                           | /bin/ssh               |
| conf.locallogins        | This parameter is only necessary when authentication is being performed for a port. When set to one, it is possible to log in to the AlterPath Console Server directly by placing a “!” before your login name, then using your normal password. This is useful if the Radius authentication server is down.           | 0                      |
| all.host                | The IP address of the host to which the terminals will connect.                                                                                                                                                                                                                                                        | 200.200.200.3          |
| all.term                | This parameter defines the terminal type assumed when performing rlogin or telnet to other hosts.                                                                                                                                                                                                                      | vt100                  |
| all.userauto            | Username used when connected to a UNIX server from the user’s serial terminal.                                                                                                                                                                                                                                         |                        |
| all.dcd<br>(CAS and TS) | See description in CAS section.                                                                                                                                                                                                                                                                                        |                        |
| all.protocol (for TS)   | For the terminal server configuration, the possible protocols are login (which requests username and password), rlogin (receives username from the (A)CS and requests a password), telnet, ssh, ssh2, or socket_client. See all.socket_port definition to see when all.protocol should be configured as socket_client. | rlogin                 |
| all.issue (CAS and TS)  | See description in CAS section.                                                                                                                                                                                                                                                                                        |                        |

# Appendix C - The pslave Configuration File

Table 29: TS Parameters

| Parameter               | Description                                                                                                                                                   | Value for this Example |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| all.prompt (CAS and TS) | See description in CAS section.                                                                                                                               |                        |
| all.socket_port         | The socket_port is the TCP port number of the application that will accept connection requested by this serial port. That application usually is telnet (23). |                        |
| s16.tty (TS)            | See the s1.tty entry in the CAS section.                                                                                                                      | ttyS16                 |

## Dial-in Access Parameters

The following parameters are unique to a Dial-in setup except where indicated.

Table 30: Dial-in configuration Parameters

| Parameter                       | Description                             | Value for this Example |
|---------------------------------|-----------------------------------------|------------------------|
| conf.pppd                       | Location of the ppp daemon with Radius. | /usr/local/sbin/pppd   |
| conf.facility (CAS and Dial-in) | See description in CAS section.         |                        |
| all.ipno (CAS and Dial-in)      | See description in CAS section.         |                        |

# Appendix C - The pslave Configuration File

Table 30: Dial-in configuration Parameters

| Parameter    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Value for this Example                                                                                                                                                                                |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| all.initchat | Modem initialization string.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <pre>TIMEOUT 10 "" \d\\dATZ \ OK\r\n-ATZ-OK\r\n "" \ "" ATMO OK\R\n "" \ TIMEOUT 3600 RING "" \ STATUS Incoming %p:I.HANDSHAKE "" ATA\ TIMEOUT 60 CONNECT@ "" \ STATUS Connected %p:I.HANDSHAKE</pre> |
| all.autoppp  | all.autoppp PPP options to auto-detect a ppp session. The cb-script parameter defines the file used for callback and enables negotiation with the callback server. Callback is available in combination with Radius Server authentication. When a registered user calls the (A)CS, it will disconnect the user, then call the user back. The following three parameters must be configured in the Radius Server: attribute Service_type(6): Callback Framed; attribute Framed_Protocol(7): PPP; attribute Callback_Number(19): the dial number (example: 50903300). | <pre>%j novj \ proxyarp modem asyncmap 000A0000 \ noipx noccp login auth require-pap refusechap\ mtu %t mru %t \ cb-script /etc/portslave/cb_script \ plugin /usr/lib/libpsr.so</pre>                 |
| all.pppopt   | all.pppopt PPP options when user has already been authenticated.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <pre>%i:%j novj \ proxyarp modem asyncmap 000A0000 \ noipx noccp mtu %t mru %t netmask%m \ idle %I maxconnect %T \ plugin /usr/lib/libpsr.so</pre>                                                    |

# Appendix C - The pslave Configuration File

---

---

Table 30: Dial-in configuration Parameters

| Parameter    | Description                                                                     | Value for this Example |
|--------------|---------------------------------------------------------------------------------|------------------------|
| all.protocol | For the Dial-in configuration, the available protocols are PPP, SLIP and CSLIP. | ppp                    |
| s32.tty      | See the s1.tty entry in the CAS section.                                        | ttyS32                 |

# Appendix D - Linux-PAM

---

---

## Introduction

Linux-PAM (Pluggable Authentication Modules for Linux) is a suite of shared libraries that enable the local system administrator to choose how applications authenticate users. In other words, without (rewriting and) recompiling a PAM-aware application, it is possible to switch between the authentication mechanism(s) it uses. Indeed, one may entirely upgrade the local authentication system without touching the applications themselves.

It is the purpose of the Linux-PAM project to separate the development of privilege-granting software from the development of secure and appropriate authentication schemes. This is accomplished by providing a library of functions that an application may use to request that a user be authenticated. This PAM library is configured locally with a system file, `/etc/pam.conf` (or a series of configuration files located in `/etc/pam.d/`) to authenticate a user request via the locally available authentication modules. The modules themselves will usually be located in the directory `/lib/security` and take the form of dynamically loadable object files.

The Linux-PAM authentication mechanism gives to the system administrator the freedom to stipulate which authentication scheme is to be used. S/he has the freedom to set the scheme for any/all PAM-aware applications on your Linux system. That is, s/he can authenticate from anything as generous as simple trust (`pam_permit`) to something as severe as a combination of a retinal scan, a voice print and a one-time password!

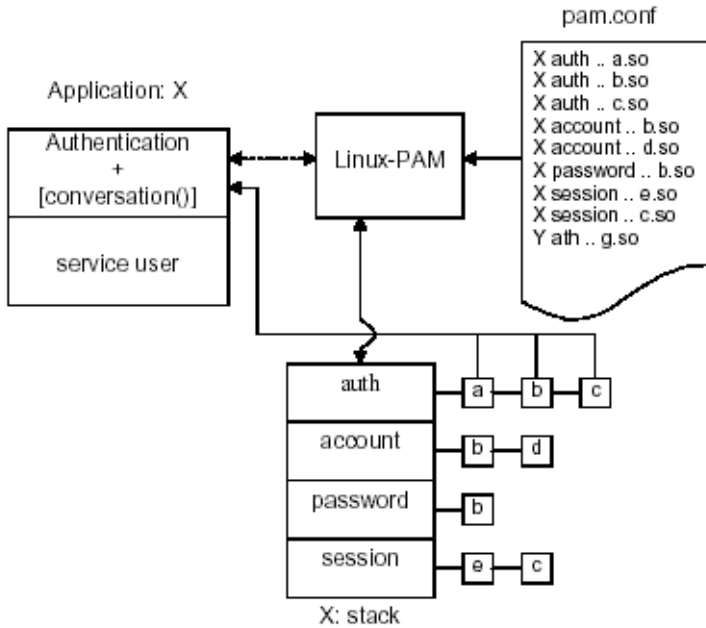
Linux-PAM deals with four separate types of (management) task. These are: authentication management, account management, session management, and password management. The association of the preferred management scheme with the behavior of an application is made with entries in the relevant Linux-PAM configuration file. The management functions are performed by modules specified in the configuration file.

Following is a figure that describes the overall organization of Linux-PAM:

# Appendix D - Linux-PAM

---

---



*Figure 43: Data flow diagram of Linux-PAM*

The left of the figure represents the application: Application X. Such an application interfaces with the Linux-PAM library and knows none of the specifics of its configured authentication method. The Linux-PAM library (in the center) consults the contents of the PAM configuration file and loads the modules that are appropriate for Application X. These modules fall into one of four management groups (lower center) and are stacked in the order they appear in the configuration file. These modules, when called by Linux-PAM, perform the various authentication tasks for the application. Textual information, required from or offered to the user can be exchanged through the use of the application-supplied conversation function.



# Appendix D - Linux-PAM

---

## The Linux-PAM Configuration File

Linux-PAM is designed to provide the system administrator with a great deal of flexibility in configuring the privilege-granting applications of their system. The local configuration of those aspects of system security controlled by Linux-PAM is contained in one of two places: either the single system file `/etc/pam.conf` or the `/etc/pam.d/` directory. In this section we discuss the correct syntax of and generic options respected by entries to these files.

### Configuration File Syntax

The reader should note that the Linux-PAM-specific tokens in this file are case-insensitive. The module paths, however, are case-sensitive since they indicate a file's name and reflect the case-dependence of typical Linux file systems. The case-sensitivity of the arguments to any given module is defined for each module in turn.

In addition to the lines described below, there are two special characters provided for the convenience of the system administrator: comments are preceded by a '#' and extend to the next end-of-line. Module specification lines may be extended with a '\\' escaped new-line.

A general configuration line of the `/etc/pam.conf` file has the following form:

```
Service-name module-type control-flag module-path arguments
```

The meaning of each of these tokens is explained below. The second (and more recently adopted) way of configuring Linux-PAM is via the contents of the `/etc/pam.d/` directory. After the meaning of the above tokens is explained, the method will be described.

*Service-name*      The name of the service associated with this entry. Frequently the service name is the conventional name of the given application. For example, 'ftpd', 'rlogind', 'su', etc. There is a special service-name, reserved for defining a default authentication mechanism. It has the name 'OTHER' and may be specified in either lower or upper case characters. Note, when there is a module specified for a named service, the 'OTHER' entries are ignored.

*Module-type*      One of (currently) the four types of module. The four types are as follows:

# Appendix D - Linux-PAM

---

---

*Auth*- This module type provides two aspects of authenticating the user. First, it establishes that the user is who they claim to be, by instructing the application to prompt the user for a password or other means of identification. Second, the module can grant group membership, independently of the `/etc/groups`, or other privileges through its credential-granting properties.

*Account*- This module performs non-authentication-based account management. It is typically used to restrict or permit access to a service based on the time of day, currently available system resources (maximum number of users) or perhaps the location of the applicant user—'root' login only on the console.

*Session*- Primarily, this module is associated with doing things that need to be done for the user before or after they can be given service. Such things include the logging of information concerning the opening or closing of some data exchange with a user, mounting directories, etc.

*Password*- This last module type is required for updating the authentication token associated with the user. Typically, there is one module for each 'challenge/response' based authentication (auth) module-type.

## *Control-flag*

The control-flag is used to indicate how the PAM library will react to the success or failure of the module it is associated with. Since modules can be stacked (modules of the same type execute in series, one after another), the control-flags determine the relative importance of each module. The application is not made aware of the individual success or failure of modules listed in the `/etc/pam.conf` file. Instead, it receives a summary of success or fail responses from the Linux-PAM library. The order of execution of these modules is that of the entries in the `/etc/pam.conf` file: earlier entries are executed before later ones. The control-flag can be defined with one of two syntaxes. The simpler (and historical) syntax for the control-flag is a single keyword defined to indicate the severity of concern associated with the success or failure of a specific module. There are four such keywords: required, requisite, sufficient and optional.

# Appendix D - Linux-PAM

---

The Linux-PAM library interprets these keywords in the following manner:

- Required* This indicates that the success of the module is required for the module-type facility to succeed. Failure of this module will not be apparent to the user until all of the remaining modules (of the same module-type) have been executed.
- Requisite* This is similar to *required*. However, in the case that such a module returns a failure, control is directly returned to the application. The return value is that associated with the first required or requisite module to fail. Note that this flag can be used to protect against the possibility of a user getting the opportunity to enter a password over an unsafe medium. It is conceivable that such behavior might inform an attacker of valid accounts on a system. This possibility should be weighed against the significant concerns of exposing a sensitive password in a hostile environment.
- Sufficient* The success of this module is deemed 'sufficient' to satisfy the Linux-PAM library that this moduletype has succeeded in its purpose. In the event that no previous required module has failed, no more 'stacked' modules of this type are invoked. (Note: in this case subsequent required modules are not invoked.) A failure of this module is not deemed as fatal to satisfying the application.
- Optional* As its name suggests, this control-flag marks the module as not being critical to the success or failure of the user's application for service. In general, Linux-PAM ignores such a module when determining if the module stack will succeed or fail. However, in the absence of any definite successes or failures of previous or subsequent stacked modules this module will determine the nature of the response to the application. One example of this latter case is when the other modules return something like PAM\_IGNORE.

# Appendix D - Linux-PAM

---

---

## Newest Syntax

The more elaborate (newer) syntax is much more specific and gives the administrator a great deal of control over how the user is authenticated. This form of the control flag is delimited with square brackets and consists of a series of value=action tokens:

```
[value1=action1 value2=action2 ...]
```

Here, value1 is one of the following return values: success; open\_err; symbol\_err; service\_err; system\_err; buf\_err; perm\_denied; auth\_err; cred\_insufficient; authinfo\_unavail; user\_unknown; maxtries; new\_authtok\_reqd; acct\_expired; session\_err; cred\_unavail; cred\_expired; cred\_err; no\_module\_data; conv\_err; authtok\_err; authtok\_recover\_err; authtok\_lock\_busy; authtok\_disable\_aging; try\_again; ignore; abort; authtok\_expired; module\_unknown; bad\_item; and default. The last of these (default) can be used to set the action for those return values that are not explicitly defined.

The action can be a positive integer or one of the following tokens: ignore, ok, done, bad, die, and reset.

- |                           |                                                                                                                                                                                                                                                                                                                                        |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>A positive integer</i> | When specified as the action, can be used to indicate that the next J modules of the current type will be skipped. In this way, the administrator can develop a moderately sophisticated stack of modules with a number of different paths of execution. Which path is taken can be determined by the reactions of individual modules. |
| <i>Ignore</i>             | When used with a stack of modules, the module's return status will not contribute to the return code the application obtains.                                                                                                                                                                                                          |
| <i>Bad</i>                | This action indicates that the return code should be thought of as indicative of the module failing. If this module is the first in the stack to fail, its status value will be used for that of the whole stack.                                                                                                                      |
| <i>Die</i>                | Equivalent to <i>bad</i> with the side effect of terminating the module stack and PAM immediately returning to the application.                                                                                                                                                                                                        |

# Appendix D - Linux-PAM

---

|              |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>OK</i>    | This tells PAM that the administrator thinks this return code should contribute directly to the return code of the full stack of modules. In other words, if the former state of the stack would lead to a return of PAM_SUCCESS, the module's return code will override this value. Note: if the former state of the stack holds some value that is indicative of a module failure, this 'OK' value will not be used to override that value. |
| <i>Done</i>  | Equivalent to OK with the side-effect of terminating the module stack and PAM immediately returning to the application.                                                                                                                                                                                                                                                                                                                       |
| <i>Reset</i> | Clear all memory of the state of the module stack and start again with the next stacked module.                                                                                                                                                                                                                                                                                                                                               |

## Module Path

Module Path is the path-name of the dynamically loadable object file—the pluggable module itself. If the first character of the module path is '/', it is assumed to be a complete path. If this is not the case, the given module path is appended to the default module path: /lib/security.

Currently, the AlterPath Console Server has the following modules available:

|                   |                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>pam_access</i> | Provides logdaemon style login access control.                                                                                                                                                                                                                                                                                                                                                           |
| <i>pam_deny</i>   | Deny access to all users.                                                                                                                                                                                                                                                                                                                                                                                |
| <i>pam_env</i>    | This module allows the (un)setting of environment variables. The use of previously set environment variables as well as PAM_ITEMS such as PAM_RHOST is supported.                                                                                                                                                                                                                                        |
| <i>pam_filter</i> | This module was written to offer a plug-in alternative to programs like ttysnoop (XXX - need a reference). Since a filter that performs this function has not been written, it is currently only a toy. The single filter provided with the module simply transposes upper and lower case letters in the input and output streams. (This can be very annoying and is not kind to termcap-based editors.) |

# Appendix D - Linux-PAM

---

---

|                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>pam_group</i>     | This module provides group settings based on the user's name and the terminal they are requesting a given service from. It takes note of the time of day.                                                                                                                                                                                                                                                                                                                           |
| <i>pam_issue</i>     | This module presents the issue file (/etc/issue by default) when prompting for a username.                                                                                                                                                                                                                                                                                                                                                                                          |
| <i>pam_lastlog</i>   | This session module maintains the /var/log/lastlog file. It adds an open entry when called via the pam_open_session() function and completes it when pam_close_session() is called. This module can also display a line of information about the last login of the user. If an application already performs these tasks, it is not necessary to use this module.                                                                                                                    |
| <i>pam_limits</i>    | This module, through the Linux-PAM open-session hook, sets limits on the system resources that can be obtained in a user session. Its actions are dictated more explicitly through the configuration file discussed below.                                                                                                                                                                                                                                                          |
| <i>pam_listfile</i>  | The listfile module provides a way to deny or allow services based on an arbitrary file.                                                                                                                                                                                                                                                                                                                                                                                            |
| <i>pam_motd</i>      | This module outputs the motd file (/etc/motd by default) upon successful login.                                                                                                                                                                                                                                                                                                                                                                                                     |
| <i>pam_nologin</i>   | Provides standard Unix nologin authentication.                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <i>pam_permit</i>    | This module should be used with extreme caution. Its action is to always permit access. It does nothing else.                                                                                                                                                                                                                                                                                                                                                                       |
| <i>pam_radius</i>    | Provides Radius server authentication and accounting.                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <i>pam_rootok</i>    | This module is for use in situations where the superuser wishes to gain access to a service without having to enter a password.                                                                                                                                                                                                                                                                                                                                                     |
| <i>pam_securetty</i> | Provides standard UNIX securetty checking.                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <i>pam_time</i>      | Running a well-regulated system occasionally involves restricting access to certain services in a selective manner. This module offers some time control for access to services offered by a system. Its actions are determined with a configuration file. This module can be configured to deny access to (individual) users based on their name, the time of day, the day of week, the service they are applying for and their terminal from which they are making their request. |

# Appendix D - Linux-PAM

---

|                    |                                                                                                                                                                                                                                                                                                        |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>pam_tacplus</i> | Provides TacacsPlus Server authentication, authorization (account management), and accounting (session management).                                                                                                                                                                                    |
| <i>pam_unix</i>    | This is the standard UNIX authentication module. It uses standard calls from the system's libraries to retrieve and set account information as well as authentication. Usually this is obtained from the <code>etc/passwd</code> and the <code>/etc/shadow</code> file as well when shadow is enabled. |
| <i>pam_warn</i>    | This module is principally for logging information about a proposed authentication or application to update a password.                                                                                                                                                                                |

## PAM Kerberos

The `pam_kerberos` module currently used is `pam_krb5`.

This PAM module requires the MIT 1.1+ release of Kerberos, or the Cygnus CNS distribution. It has not been tested against heimdal or any other Kerberos distributions.

### Important Files

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>/etc/krb5.conf</i> | The <code>krb5.conf</code> file contains Kerberos configuration information, including the locations of KDCs and admin servers for the Kerberos realms of interest, defaults for the current realm and for Kerberos applications, and mappings of hostnames onto Kerberos realms. Normally, you should install your <code>krb5.conf</code> file in the <code>directory/etc</code> . You can override the default location by setting the environment variable <code>KRB5_CONFIG</code> . |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## PAM LDAP

`Pam_ldap` looks for the `ldap` client configuration file "`ldap.conf`" in `/etc/`. Here's an example of the `ldap.conf` file (partial):

```
file name: ldap.conf
This is the configuration file for the LDAP nameservice
switch library and the LDAP PAM module.
#
```

# Appendix D - Linux-PAM

---

---

```
Your LDAP server. Must be resolvable without using LDAP.
host 127.0.0.1

The distinguished name of the search base.
base dc=padl,dc=com
```

## Arguments

The arguments are a list of tokens that are passed to the module when it is invoked. They are much like arguments to a typical Linux shell command. Generally, valid arguments are optional and are specific to any given module. Invalid arguments are ignored by a module, however, when encountering an invalid argument, the module is required to write an error to `syslog(3)`.

The following are optional arguments which are likely to be understood by any module. Arguments (including these) are in general, optional.

|                       |                                                                                                                                                                                                                                                                                            |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>debug</i>          | Use the <code>syslog(3)</code> call to log debugging information to the system log files.                                                                                                                                                                                                  |
| <i>no_warn</i>        | Instruct module to not give warning messages to the application.                                                                                                                                                                                                                           |
| <i>use_first_pass</i> | The module should not prompt the user for a password. Instead, it should obtain the previously typed password (from the preceding auth module), and use that. If that doesn't work, then the user will not be authenticated. (This option is intended for auth and password modules only). |
| <i>try_first_pass</i> | The module should attempt authentication with the previously typed password (from the preceding auth module). If that doesn't work, then the user is prompted for a password. (This option is intended for auth modules only).                                                             |



# Appendix D - Linux-PAM

---

*use\_mapped\_pass* This argument is not currently supported by any of the modules in the Linux-PAM distribution because of possible consequences associated with U.S. encryption exporting restrictions.

*expose\_account* In general, the leakage of some information about user accounts is not a secure policy for modules to adopt. Sometimes information such as user names or home directories, or preferred shell, can be used to attack a user's account. In some circumstances, however, this sort of information is not deemed a threat: displaying a user's full name when asking them for a password in a secured environment could also be called being 'friendly'. The *expose\_account* argument is a standard module argument to encourage a module to be less discrete about account information as deemed appropriate by the local administrator. Any line in (one of) the configuration file(s), that is not formatted correctly will generally tend (erring on the side of caution) to make the authentication process fail. A corresponding error is written to the system log files with a call to `syslog(3)`.

## Directory-based Configuration

It is possible to configure `libpam` via the contents of the `/etc/pam.d/` directory. This is more flexible than using the single configuration file. In this case, the directory is filled with files—each of which has a filename equal to a service-name (in lower-case)—the personal configuration file for the named service. The AlterPath Console Server Linux-PAM was compiled to use both `/etc/pam.d/` and `/etc/pam.conf` in sequence. In this mode, entries in `/etc/pam.d/` override those of `/etc/pam.conf`.

The syntax of each file in `/etc/pam.d/` is similar to that of the `/etc/pam.conf` file and is made up of lines of the following form:

```
module-type control-flag module-path arguments
```

The only difference between the two is that the service-name is not present. The service-name is of course the name of the given configuration file. For example, `/etc/pam.d/login` contains the configuration for the login service.

# Appendix D - Linux-PAM

---

---

## Default Policy

If a system is to be considered secure, it had better have a reasonably secure ‘OTHER’ entry. The following is a “severe” setting (which is not a bad place to start!):

```
#
default; deny access
#
OTHER auth required pam_deny.so
OTHER account required pam_deny.so
OTHER password required pam_deny.so
OTHER session required pam_deny.so
```

While fundamentally a secure default, this is not very sympathetic to a misconfigured system. For example, such a system is vulnerable to locking everyone out should the rest of the file become badly written.

The module `pam_deny` not very sophisticated. For example, it logs no information when it is invoked, so unless the users of a system contact the administrator when failing to execute a service application, the administrator may not know for a long while that his system is misconfigured.

The addition of the following line before those in the above example would provide a suitable warning to the administrator.

```
#
default; wake up! This application is not configured
#
OTHER auth required pam_warn.so
OTHER password required pam_warn.so
```

Having two “OTHER auth” lines is an example of stacking.

# Appendix D - Linux-PAM

---

On a system that uses the `/etc/pam.d/` configuration, the corresponding default setup would be achieved with the following file:

```
#
default configuration: /etc/pam.d/other
#
auth required pam_warn.so
auth required pam_deny.so
account required pam_deny.so
password required pam_warn.so
password required pam_deny.so
session required pam_deny.so
```

On a less sensitive computer, the following selection of lines (in `/etc/pam.conf`) is likely to mimic the historically familiar Linux setup:

```
#
default; standard UNIX access
#
OTHER auth required pam_unix_auth.so
OTHER account required pam_unix_acct.so
OTHER password required pam_unix_passwd.so
OTHER session required pam_unix_session.so
```

In general this will provide a starting place for most applications.

In addition to the normal applications: `login`, `su`, `sshd`, `passwd`, and `pppd`. Cyclades also has made `portslave` a PAM-aware application. The `portslave` requires four services configured in `pam.conf`. They are `local`, `remote`, `radius`, and `tacplus`. The `portslave` PAM interface takes any parameter needed to perform the authentication in the serial ports from the file `pslave.conf`. The `pslave.conf` parameter `all.auththtype` determines which service(s)

# Appendix D - Linux-PAM

---

---

should be used.

```
-----#
/etc/pam.conf
#
#
#
Last modified by Andrew G. Morgan <morgan@kernel.org>
#
-----#
$Id: pam.conf,v 1.2 2001/04/08 06:02:33 agmorgan Exp $
-----#
serv. module ctrl module [path] ...[args..]
#
name type flag
#
-----#
#
The PAM configuration file for the 'tacplus' service
#
tacplus auth requisite pam_securetty.so
tacplus auth required pam_tacplus.so encrypt
tacplus account required pam_tacplus.so encrypt service=ppp proto-
col=lcp
tacplus session required pam_tacplus.so encrypt service=ppp proto-
col=lcp
#
```

# Appendix D - Linux-PAM

---

```
The PAM configuration file for the 'radius' service
#
radius auth requisite pam_securetty.so
radius auth required pam_radius_auth.so
radius account required pam_radius_auth.so
radius session required pam_radius_auth.so
#
The PAM configuration file for the 'local' service
#
local auth requisite pam_securetty.so
local auth required pam_unix.so
local account required pam_unix.so
local password required pam_unix.so md5 use_authtok
local session required pam_unix.so
#
The PAM configuration file for the 'remote' service
#
remote auth required pam_permit.so
remote account required pam_permit.so
remote password required pam_permit.so
remote session required pam_permit.so
#
The PAM configuration file for the 'login' service
#
```

# Appendix D - Linux-PAM

---

---

```
login auth requisite pam_securetty.so
login auth required pam_unix.so
login auth optional pam_group.so
login account requisite pam_time.so
login account required pam_unix.so
login password required pam_unix.so md5 use_authtok
login session required pam_unix.so
login session required pam_limits.so
#
The PAM configuration file for the 'xsh' service
#
sshd auth requisite pam_securetty.so
sshd auth required pam_unix.so
sshd auth optional pam_group.so
sshd account requisite pam_time.so
sshd account required pam_unix.so
sshd password required pam_unix.so md5 use_authtok
sshd session required pam_unix.so
sshd session required pam_limits.so
#
The PAM configuration file for the 'passwd' service
#
passwd password required pam_unix.so md5
#
```

# Appendix D - Linux-PAM

---

```
The PAM configuration file for the 'samba' service
#
samba auth required pam_unix.so
samba account required pam_unix.so
#
The PAM configuration file for the 'su' service
#
su auth required pam_wheel.so
su auth sufficient pam_rootok.so
su auth required pam_unix.so
su account required pam_unix.so
su session required pam_unix.so
#
Information for the PPPD process with the 'login' option.
#
ppp auth required pam_nologin.so
ppp auth required pam_unix.so
ppp account required pam_unix.so
ppp session required pam_unix.so
#
The PAM configuration file for the 'other' service
#
other auth required pam_warn.so
other auth required pam_deny.so
```

# Appendix D - Linux-PAM

---

---

```
other account required pam_deny.so
other password required pam_warn.so
other password required pam_deny.so
other session required pam_deny.so
```

## Reference

**The Linux-PAM System Administrators' Guide**  
Copyright (c) Andrew G. Morgan 1996-9. All rights reserved.  
Email: [morgan@linux.kernel.org](mailto:morgan@linux.kernel.org)



# Appendix E - Customization and the CDK

---

## Introduction

Everything related to the AlterPath Console Server can be traced back to two files:

- /etc/rc.sysinit
- /etc/inittab

All AlterPath Console Server application programs are started during boot by the init process. The related lines in the /etc/inittab file are listed below:

```
System initialization.
::sysinit:/etc/rc.sysinit

Single user shell

#console::respawn:/bin/sh < /dev/console > /dev/console 2> /dev/
console

ttyS0::respawn:/sbin/getty -p ttyS0 ansi
::respawn:/sbin/cy_wdt_led wdt led

Cyclades RAS
::once:/sbin/cron
::once:/sbin/snmpd
::once:/sbin/cy_buffering
::once:/sbin/cy_ras
::once:/sbin/sshd -f /etc/ssh/sshd_config
::once:/sbin/ex_ntpclient
::once:/bin/webs
::once:/bin/syslog-ng
::once:/bin/cy_alarm
::wait:/sbin/fwset restore
```

# Appendix E - Customization and the CDK

---

---

## The Customization Process

To customize the AlterPath Console Server, change these lines or add others. If the `/etc/inittab` file is changed, edit the `/etc/config_files` file and add a line containing only `"/etc/inittab"`. Save the file and exit the editor. Save the new configuration by executing `saveconf`. Then, the AlterPath Console Server should be rebooted. This is necessary because the `init` program provided by `Busybox`--a tool that emulates `rm`, `cp`, etc., but uses much less space--does not support the option `q`.

## The Cyclades Development Kit

Cyclades provides a development kit which allows changes to be made to the AlterPath Console Server's software. However, Cyclades does not provide free technical support for systems modified in this way. Any changes are the responsibility of the user. The Cyclades Developer Kit (CDK) is available on the Cyclades Web site. Contact Tech Support to download the CDK.

# Appendix F - Upgrades and Troubleshooting

---

## Upgrades

Users should upgrade the AlterPath Console Server whenever there is a bug fix or new features that they would like to have. Below are the six files added by Cyclades to the standard Linux files in the /proc/flash directory when an upgrade is needed. They are:

- `boot_ori` - original boot code
- `boot_alt` - alternate boot code
- `syslog` - event logs (not used by Linux)
- `config` - configuration parameters, only the boot parameters are used by the boot code
- `zImage` - Linux kernel image
- `script` - file where all AlterPath Console Server configuration information is stored

### The Upgrade Process

To upgrade the AlterPath Console Server, follow these steps:

**Step 1:** Log in to the (A)CS as root.

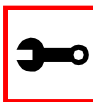
Provide the root password if requested.

**Step 2:** Go to the /proc/flash directory using the following command:

```
cd /proc/flash
```

**Step 3:** FTP to the host where the new firmware is located.

Log in using your username and password. Go to the directory where the firmware is located. Select binary transfer and “get” the firmware file.



**Note:** The destination file name in the /proc/flash directory must be `zImage`.  
Example (hostname = server; directory = /tftpboot; username= admin;  
password = adminpw; firmware filename on that server = `zImage.210`).

# Appendix F - Upgrades and Troubleshooting

---

```
ftp
> open server
> user admin
> Password: adminpw
> cd /tftpboot
> bin
> get zImage.210 zImage
> quit
```



**Note:** Due to space limitations, the new zImage file may not be downloaded with a different name, then renamed. The (A)CS searches for a file named zImage when booting and there is no room in flash for two zImage files.

## Step 4: Run zImage.

To make sure the downloaded file is not corrupted or that the zImage saved in flash is OK the user should run:

```
md5sum -b /proc/flash/zImage
```

## Step 5: Check text file information.

Now the user should check with the information present in the text file saved in the Cyclades site (e.g. zImage.210.md5sum). If the numbers match, the downloaded file is not corrupted.

## Step 6: Issue the command reboot.

```
reboot
```

## Step 7: Confirm that the new Linux kernel has taken over.

After rebooting, the new Linux kernel will take over. This can be confirmed by typing

```
cat /proc/version
```

to see the Linux kernel version.

# Appendix F - Upgrades and Troubleshooting

---

## Troubleshooting

### Flash Memory Loss

If the contents of flash memory are lost after an upgrade, please follow the instructions below to restore your system:

**Step 1:** Turn the (A)CS OFF, then back ON.

**Step 2:** Using the console, during the self test, press <Esc> after the Ethernet test.

**Step 3:** When the Watch Dog Timer prompt appears, press <Enter>.

**Step 4:** Choose the option Network Boot when asked.

**Step 5:** Enter the IP address of the Ethernet interface.

**Step 6:** Enter the IP address of the host where the new zImage file is located.

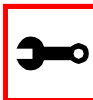
**Step 7:** Enter the file name of the zImage file on the host.

**Step 8:** Select the TFTP option instead of BOOTP.

The host must be running TFTPD and the new zImage file must be located in the proper directory. e.g. /tftpboot for Linux.

**Step 9:** Accept the default MAC address by pressing <Enter>.

The (A)CS should begin to boot off the network and the new image will be downloaded and begin running in RAM. At this point, follow the upgrade steps above (login, cd /proc/flash, ftp, and so forth) to save the new zImage file into flash again.



**Note:** Possible causes for the loss of flash memory may include: downloaded wrong zImage file, downloaded as ASCII instead of binary; problems with flash memory.

# Appendix F - Upgrades and Troubleshooting

---

---

If the AlterPath Console Server booted properly, the interfaces can be verified using *ifconfig* and *ping*. If ping does not work, check the routing table using the command *route*. Of course, all this should be tried after checking that the cables are connected correctly.

The file */etc/config\_files* contains a list of files acted upon by *saveconf* and *restoreconf*. If a file is missing, it will not be loaded onto the ramdisk on boot. The following table lists files that should be included in the */etc/config\_files* file and which programs use each.

Table 31: Files to be included in */etc/config\_file* and the program to use

| File                             | Program                   |
|----------------------------------|---------------------------|
| <i>/etc/securetty</i>            | telnet, login, su         |
| <i>/etc/issue</i>                | getty                     |
| <i>/etc/getty_ttyS0</i>          | login (via console)       |
| <i>/etc/hostname</i>             | tcp                       |
| <i>/etc/hosts</i>                | tcp                       |
| <i>/etc/host.conf</i>            | tcp                       |
| <i>/etc/nsswitch.conf</i>        | dns                       |
| <i>/etc/resolv.conf</i>          | dns                       |
| <i>/etc/config_files</i>         | saveconf                  |
| <i>/etc/passwd</i>               | login, passwd, adduser... |
| <i>/etc/group</i>                | login, passwd, adduser... |
| <i>/etc/ssh/ssh_host_key.pub</i> | sshd                      |
| <i>/etc/ssh/sshd_config</i>      | sshd                      |
| <i>/etc/ssh/ssh_config</i>       | ssh client                |
| <i>/etc/ssh/ssh_host_key</i>     | sshd (ssh1)               |

# Appendix F - Upgrades and Troubleshooting

Table 31: Files to be included in /etc/config\_file and the program to use

| File                                  | Program                                            |
|---------------------------------------|----------------------------------------------------|
| <i>/etc/ssh/ssh_host_key.pub</i>      | sshd (ssh1)                                        |
| <i>/etc/ssh/ssh_host_dsa_key</i>      | sshd (ssh2)                                        |
| <i>/etc/ssh/ssh_host_dsa_key.pub</i>  | sshd (ssh2)                                        |
| <i>/etc/snmp/snmpd.conf</i>           | snmpd                                              |
| <i>/etc/portslave/pslave.conf</i>     | cy_ras, portslave, (A)CS configuration information |
| <i>/etc/network/ifcfg_eth0</i>        | ifconfig eth0, cy_ras, rc.sysinit                  |
| <i>/etc/network/ifcfg*</i>            | ifconfig, cy_ras, rc.sysinit                       |
| <i>/etc/network/ifcfg_lo ifconfig</i> | lo, cy_ras, rc.sysinit                             |
| <i>/var/run/radsession.id</i>         | radinit, radius authentication process             |
| <i>/home</i>                          | adduser, passwd                                    |
| <i>/etc/network/st_routes</i>         | ifconfig, cy_ras, rc.sysinit                       |
| <i>/etc/syslog-ng/syslog-ng.conf</i>  | syslog-ng                                          |



**Important!** If any of the files listed in /etc/config\_files is modified, the AlterPath Console Server administrator must execute the command *saveconf* before rebooting the AlterPath Console Server or the changes will be lost. If a file is created (or a filename altered), its name must be added to this file before executing *saveconf* and rebooting.

# Appendix F - Upgrades and Troubleshooting

---

---



**Important!** Cyclades Technical Support is always ready to help with any configuration problems. Before calling, execute the command

```
cat /proc/version
```

and note the Linux version and AlterPath Console Server version written to the screen. This will speed the resolution of most problems.

## Hardware Test

A hardware test called *tstest* is included with the AlterPath Console Server firmware. It is a menu-driven program, run by typing *tstest* at the command prompt. The various options are described below. Note that the AlterPath Console Server should not be tested while in use as the test will inactivate all ports. You should inactivate all processes that may use the serial ports: *inetd*, *sshd*, *cy\_ras*, and *cy\_buffering*. Following are the hardware test steps:

Step 1: *signal\_ras* stop.

Step 2: Perform all hardware tests needed.

Step 3: *signal\_ras* start.

## Port Test

Either a cross cable or a loop-back connector is necessary for this test. Their pinout diagrams are supplied in [Appendix B - Cabling, Hardware, and Electrical Specifications](#). Connect the loop-back connector to the modem cable and then connect the modem cable to the port to be tested (or connect a cross cable between two ports to be tested). When *tstest* senses the presence of the cable or connector, the test will be run automatically and the result shown on the screen.

Each line of data corresponds to a port in test. The last four columns (DATA, CTS, DCD, and DSR) indicate errors. The values in these columns should be zero. Below is an example of the output screen.



# Appendix F - Upgrades and Troubleshooting

---

|      |       | <- Packets -> |          | <- Errors -> |      |     |     |     |
|------|-------|---------------|----------|--------------|------|-----|-----|-----|
| From | To    | Sent          | Received | Passes       | Data | CTS | DCD | DSR |
| 2    | <-> 2 | 35            | 35       | 35           | 0    | 0   | 0   | 0   |
| 4    | <-> 5 | 35            | 35       | 35           | 0    | 0   | 0   | 0   |
| 5    | <-> 4 | 35            | 35       | 35           | 0    | 0   | 0   | 0   |

When this test is run with a cable or connector without the DSR signal (see the pinout diagram for the cable or connector being used), errors will appear in the DSR column. This does not indicate a problem with the port. In the example above, `tstest` perceived that a loop-back connector was attached to port 2 and that a cross cable was used to connect ports 4 and 5.

## Port Conversation

This test sends and receives data on the selected port. One way to run this test is to place a loop-back connector on the port to be tested and begin. Enter the number of the port and a baud rate (9600 is a typical value). Type some letters, and if the letters appear on the screen, the port is working. If the letters do not appear on the screen (which also occurs if the loop-back connector is removed), the port is not functioning correctly.

A second method that can be used to test the port is to connect it to a modem with a straight cable. Begin the test and type "at". The modem should respond with "OK", which will appear on the screen. Other commands can be sent to the modem or to any other serial device.

## Test Signals Manually

This test confirms that signals are being sent and received on the selected port. Neither the loop-back connector nor the cross cable are necessary. Enter the number of the port to be tested and begin the test.

| State | DTR | DCD | DSR | RTS | CTS |
|-------|-----|-----|-----|-----|-----|
| ON    | X   |     |     | X   |     |
|       | ↓   |     |     | ↓   |     |
| OFF   |     | X   | X   |     | X   |

*Figure 44: Initial test*

# Appendix F - Upgrades and Troubleshooting

---

---

First, type Ctrl-D to see the X in the DTR column move position, then type Ctrl-R to see the X in the RTS column change position. If each of the Xs moves in response to its command, the signals are being sent. Another method to test the signals is to use a loop-back connector. Enter the number of the port with the loopback connector and start the test. In this case, when Ctrl-D is typed, the Xs in the first three columns will move as shown below.

| State | DTR | DCD | DSR | RTS | CTS |
|-------|-----|-----|-----|-----|-----|
| ON    | X   | X   | X   | X   |     |
|       | ↓   | ↓   | ↓   |     |     |
| OFF   |     |     |     |     | X   |

*Figure 45: Second screen, showing changed positions*

This is because the test is receiving the DTR signal sent through the DCD and DSR pins. When Ctrl-R is typed, the Xs in the RTS and CTS columns should move together. If the Xs change position as described, the signals are being sent and received correctly.

## Single User Mode

The AlterPath Console Server has a single user mode used when:

- The name or password of the user with root privileges is lost or forgotten,
- After an upgrade or downgrade which leaves the AlterPath Console Server unstable,
- After a configuration change which leaves the AlterPath Console Server inoperative or unstable.

Type the word “single” (with a blank space before the word) during boot using a console connection. This cannot be done using a telnet or other remote connection. The initial output of the boot process is shown below.

```
Entry Point = 0x00002120
loaded at: 00002120 0000D370
relocated to: 00300020 0030B270
board data at: 003052C8 0030537C
relocated to: 002FF120 002FF1D4
```

# Appendix F - Upgrades and Troubleshooting

---

```
zimage at: 00008100 0006827E
relocated to: 00DB7000 00E1717E
initrd at: 0006827E 0024F814
relocated to: 00E18000 00FFF596
avail ram: 0030B270 00E18000
Linux/PPC load: root=/dev/ram
```

After printing “Linux/PPC load: root=/dev/ram,” the AlterPath Console Server waits approximately 10 seconds for user input. This is where the user should type “<sp>single” (spacebar, then the word “single”). When the boot process is complete, the Linux prompt will appear on the console:

```
[root@(none) /]#
```

If the password or username was forgotten, execute the following commands:

```
passwd
saveconf
reboot
```

For configuration problems, the user has two options:

**Step 1: Edit the file(s) causing the problem with vi, then execute the commands:**

```
saveconf
reboot
```

**Step 2: Reset the configuration by executing the commands:**

```
echo 0 > /proc/flash/script
reboot
```

If the problem is due to an upgrade/downgrade, a second downgrade/upgrade will be necessary to reverse the process. First, the network must be initialized in order to reach a ftp server. Execute the following script, replacing the parameters with values appropriate for

# Appendix F - Upgrades and Troubleshooting

---

---

your system. If your ftp server is on the same network as the (A)CS, the gw and mask parameters are optional.

```
config_eth0 ip 200.200.200.1 mask 255.255.255.0 gw 200.200.200.5
```

At this point, the DNS configuration (in the file `/etc/resolv.conf`) should be checked. Then, download the kernel image using the ftp command.

## Troubleshooting the Web Configuration Manager

### What to do when the initial Web page does not appear

Try pinging, telnetting, or tracerouting to the AlterPath Console Server to make sure it is reachable. If not, the problem is probably in the network or network configuration. Are the interfaces up? Are the IP addresses correct? Are filters configured which block the packets? If the AlterPath Console Server is reachable, see if the `/bin/webs` process is running by executing the command `ps`. If it is not, type `/bin/webs &` to start it. If the `/bin/webs` process is not being initialized during boot, change the file `/etc/inittab`.

### How to restore the Default Configuration of the Web Configuration Manager

This would be required only when the root password was lost or the configuration file `/etc/websum.conf` was damaged. From a console or telnet session, edit the file `/etc/config_files`. Find the reference to `/etc/websum.conf` and delete it. Save the modified `/etc/config_files` file. Execute the command `saveconf`. Reboot the system. Enter into the Web Configuration Manager with the default username and password (`root/tslinux`). Edit the file `/etc/config_files` and insert the reference to `/etc/websum.conf`.

### Using a different speed for the Serial Console

The serial console is originally configured to work at 9600 bps. If you want to change that, it is necessary to change the configuration following the steps:

**Step 1: Run bootconf.** The user will be presented with the screen:

```
Current configuration
MAC address assigned to Ethernet [00:60:2e:00:16:b9]
IP address assigned to Ethernet interface [192.168.160.10]
Watchdog timer ((A)ctive or (I)nactive) [A]
```

# Appendix F - Upgrades and Troubleshooting

---

```
Firmware boot from ((F)lash or (N)etwork) [F]
Boot type ((B)ootp,(T)ftp or Bot(H)) [T]
Boot File Name [zvmppctsbin]
Server's IP address [192.168.160.1]
Console speed [9600]
(P)erform or (S)kip Flash test [P]
(S)kip, (Q)uick or (F)ull RAM test [F]
Fast Ethernet ((A)uto Neg, (1)00 BtH, 100 Bt(F), 10 B(t)F, 10
Bt(H)) [A]
Fast Ethernet Maximum Interrupt Events [0]
```

**Type <Enter> for all fields but the Console Speed. When presented the following line:**

```
Do you confirm these changes in flash ((Y)es, (N)o (Q)uit)
[N] :
```

**Step 2: Enter Y and the changes will be saved in flash.**

**Step 3: Logout and login again to use the console at the new speed.**

## How to connect to serial ports from the browser

Depending on how the serial port is configured, connecting to a serial port will either open up a telnet or ssh connection. A serial port configured as socket\_server or raw\_data will open up a telnet connection while socket\_ssh will open up a ssh connection.

# Appendix F - Upgrades and Troubleshooting

---

---

Tested Environment

Table 32: Windows XP + JREv1.4.0\_01 or 02

|                       |         |
|-----------------------|---------|
| Internet Explorer 6.0 | Success |
| Netscape 6/6.2.3      | Success |
| Netscape 7.0          | Success |
| Mozilla 1.1           | Success |

Table 33: Redhat 7.3 + JREv1.4.0\_01 or 02

|              |         |
|--------------|---------|
| Netscape 7.0 | Success |
| Mozilla 1.1  | Success |

Requirements: Java 2 Runtime Environment (JRE) SE v1.4.0\_01 or v1.4.0\_02 installed on your PC with your browser acknowledged to use it. You can first check if the browser you are using acknowledges the Java version by:

On Windows

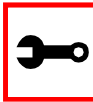
*From Internet Explorer:* Go to Tools → Internet Options → Advanced. Scroll down and look for a section on Java. There should be a checkbox that says "Use Java 2 v1.4.0 ...." If there isn't, this could either mean your browser is not activated to use the Java plug-in that came with the JRE you have installed or it just means that you don't have any JRE installed, in which case please install and repeat the check.

If you have already installed JRE and you just want to activate your browser to use it, go to your system's Control Panel → Java Plug-in icon → Browser → check on the browser(s) you want to activate to use the Java Plug-in. Now repeat the check to see if your browser will now use the correct Java Plug-in.

# Appendix F - Upgrades and Troubleshooting

---

*From Netscape or Mozilla:* Check to see if Java is enabled. Go to Edit → Preferences → Advanced → Check on Enable Java. To see what version of JRE Plug-in is used, go to Help → About Plug-ins. Scroll down to Java Plug-in section. Check if the Java Plug-in is the version you have installed.



**Tip.** When installing Netscape 7.0, it will ask if you want to install Sun Java. If you click on the box to install it, a version of JRE will be installed into your system; however, this does not mean that other browsers such as IE will recognize it. If you choose not to install Sun Java through Netscape but do it separately, Netscape 7.0 should automatically detect the JRE, and this can be checked by the instructions mentioned above.

On Linux

*From Netscape or Mozilla:* Check to see if Java is enabled. Go to Edit → Preferences → Advanced → Check on Enable Java. To see what version of JRE Plug-in is used, go to Help → About Plug-ins. Scroll down to Java Plug-in section. Check if the Java Plug-in is the version you have installed. If you have installed JRE, but the correct Java Plug-in is not shown, this may mean the browser is not locating the correct plug-in.

To fix this, go to the directory where your browser is installed. Then make a soft link from `<netscape or mozilla>/plugins/.` to the plug-in module in your JRE directory.

For example for Netscape:

```
ln -s <jre>/plugin/i386/ns600/libjavaplugin_oji.so <netscape>/plugins/.
```

where, `<jre>` is the path to your Java Runtime Environment (JRE) installation and `<netscape>` is the path to your Netscape installation. The plug-in for Mozilla should be the one under `<jre>/plugin/i386/ns610/....` After creating the link, check again to see if your browser recognized the plug-in.

# Appendix F - Upgrades and Troubleshooting

---

## Step-by-Step Process

**Step 1: Point your browser to the Console Server.**

In the address field of your browser type the IP address of your Console Server.

192.168.160.10

**Step 2: Log in.**

Log in as root, pwd is tslinux. This will take you to the Configuration and Administration page.

**Step 3: Select the Connect to Serial Ports link.**

Click on the Connect to Serial Ports link on the Link Panel to the left of the page in the Configuration section. This will take you to the Port Selection page. The ports will be listed by their serverfarm name if it were configured.



*Figure 46: Serial Port Connection page*

**Step 4: Select port.**

On the Port Selection page, choose a port to connect to from the dropdown menu and click the Submit button. This will take you to the Port Connection Page.



# Appendix F - Upgrades and Troubleshooting

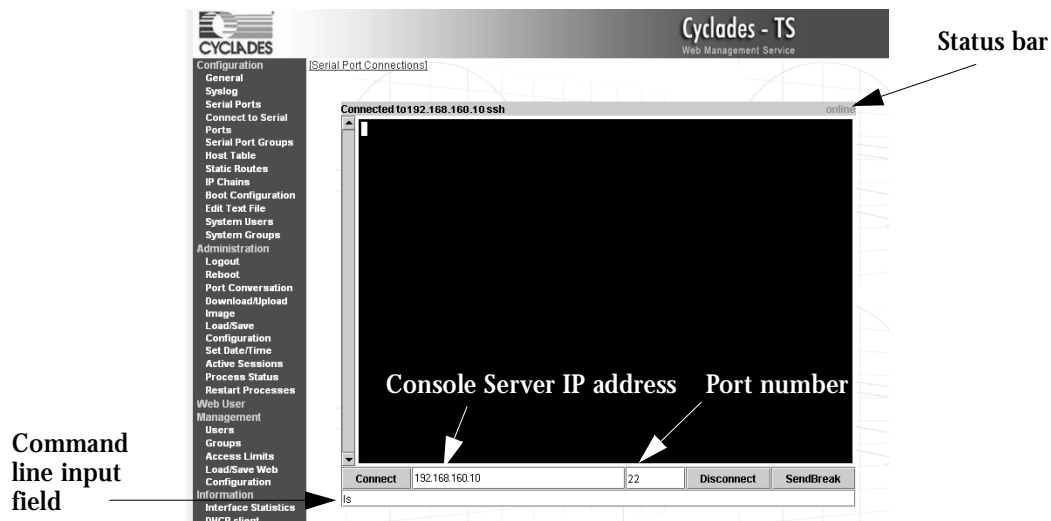


Figure 47: Port Connection page

## Step 5: Log in.

If the port selected were configured for a ssh connection, a Login window will pop up. Login in this format: <username>:<socket\_port number>. Then enter in the username's password.

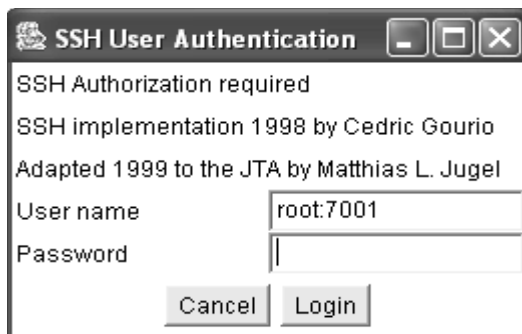


Figure 48: SSH User Authentication page

# Appendix F - Upgrades and Troubleshooting

---

If the port selected were configured as `socket_server` or `raw_data`, and depending on how it is configured to be authenticated, log in by typing into the terminal or use the command line input field to send input into the terminal.

**Step 6: Enter command.**

Enter commands directly in the terminal or into the command line input field and hit Enter.

**Step 7: To send a break to the terminal.**

Click on the SendBreak button.

**Step 8: Disconnect connection.**

Click on the Disconnect button. Make sure the Status bar shows an Offline status.

**Step 9: To reconnect to port.**

Either refresh the current page or enter in the AlterPath Console Server IP address and the port into the IP and port field. Then hit the Connect button. (For ssh connection, the port number should be 22.)

To connect to another serial port and/or AlterPath Console Server, first make sure that you have disconnected from your current session. Then enter in the IP and port number into the appropriate fields and hit Connect. If you refresh the page now, the new connection will be lost and you will be returned to the original connection.

## CPU LED

Normally the CPU status LED should blink consistently one second on, one second off. If this is not the case, an error has been detected during the boot. The blink pattern can be interpreted via the following table:

# Appendix F - Upgrades and Troubleshooting

---

---

Table 34: CPU LED Code Interpretation

| Event                              | CPU LED Morse code             |
|------------------------------------|--------------------------------|
| Normal Operation                   | S (short, short, short . . . ) |
| Flash Memory Error - Code          | L (long, long, long . . . )    |
| Flash Memory Error - Configuration | S, L                           |
| Ethernet Error                     | S, S, L                        |
| No Interface Card Detected         | S, S, S, L                     |
| Network Boot Error                 | S, S, S, S, L                  |
| Real-Time Clock Error              | S, S, S, S, S, L               |



**Note:** The Ethernet error mentioned in the above table will occur automatically if the Fast Ethernet link is not connected to an external hub during the boot. If the Fast Ethernet is not being used or is connected later, this error can be ignored.

# Appendix G - Certificate for HTTP Security

## Introduction

The following configuration will enable you to obtaining a Signed Digital Certificate. A certificate for the HTTP security is created by a CA (Certificate Authority). Certificates are most commonly obtained through *generating public and private keys*, using a public key algorithm like RSA or X509. The keys can be generated by using a key generator software.

## Procedure

Step 1: Enter OpenSSL command.

On a Linux computer, key generation can be done using the OpenSSL package, through the following command:

```
openssl req -new -nodes -keyout private.key -out public.csr
```

If this command is used, the following information is required:

Table 35: Required information for the OpenSSL package

| Parameter                                                   | Description                                                           |
|-------------------------------------------------------------|-----------------------------------------------------------------------|
| Country Name (2 letter code) [AU]:                          | The country code consisting of two letters.                           |
| State or Province Name (full name) [Some-State]:            | Provide the full name (not the code) of the state.                    |
| Locality Name (e.g., city) []:                              | Enter the name of your city.                                          |
| Organization Name (e.g., company) [Internet Widgits Ltd]:   | Organization that you work for or want to obtain the certificate for. |
| Organizational Unit Name (e.g., section) []:                | Department or section where you work.                                 |
| Common Name (e.g., your name or your server's hostname) []: | Name of the machine where the certificate must be installed.          |

# Appendix G - Certificate for HTTP Security

---

---

Table 35: Required information for the OpenSSL package

| Parameter         | Description                                              |
|-------------------|----------------------------------------------------------|
| Email Address []: | Your email address or the administrator's email address. |

The other requested information can be skipped.

The certificate signing request (CSR) generated by the command above contains some personal (or corporate) information and its public key.

## Step 2: Submit CSR to the CA.

The next step is to submit the CSR and some personal data to the CA. This service can be requested by accessing the CA Web site and is not free. There is a list of CA's at the following URL

`pki-page.org`

The request will be analyzed by the CA, for policy approval and to be signed.

## Step 3: Upon receipt, install certificate.

After the approval, the CA will send a certificate file to the origin, which we will call `Cert.cer`, for example purposes. The certificate is also stored on a directory server. The certificate must be installed in the GoAhead Web server, by following these instructions:

**Step A:** Open a Cyclades Terminal Server session and do the login.

**Step B:** Join the certificate with the private key into the file `/web/server.pem`.

```
#cat Cert.cer private.key > /web/server.pem
```

**Step C:** Copy the certificate to the file `/web/cert.pem`.

```
#cp Cert.cer /web/cert.pem
```

**Step D:** Include the files `/web/server.pem` and `/web/cert.pem` in `/etc/config_files`.

# Appendix G - Certificate for HTTP Security

---

**Step E:** Save the configuration in flash.

```
#saveconf
```

**Step F:** The certification will be effective in the next reboot.

# Appendix H - IPSEC

---

---

## Introduction

### IPsec, Security for the Internet Protocol

FreeS/WAN is a Linux implementation of the IPsec (IP security) protocols. IPsec provides encryption and authentication services at the IP (Internet Protocol) level of the network protocol stack.

Working at this level, IPsec can protect any traffic carried over IP, unlike other encryption which generally protects only a particular higher-level protocol—PGP for mail, SSH for remote login, SSL for Web work, and so on.

### Applications of IPsec

Because IPsec operates at the network layer, it is remarkably flexible and can be used to secure nearly any type of Internet traffic. Two applications, however, are extremely widespread:

- A Virtual Private Network, or VPN, allows multiple sites to communicate with the Console Server securely over an insecure Internet by encrypting all communication between the sites and the Console Server.
- "Road Warriors" connect to the Console Server from home, or perhaps from a hotel somewhere.

A somewhat more detailed description of each of these applications is below. Our Quick Start section will show you how to build each of them.

### Using secure tunnels to create a VPN

A VPN, or Virtual Private Network lets the Console Server and a whole network communicate securely when the only connection between them is over a third network which is not trustworthy. The method is to put a security gateway machine in the network and create a security tunnel between the Console Server and this gateway. The gateway machine and the Console Server encrypt packets entering the untrusted net and decrypt packets leaving it, creating a secure tunnel through it.

# Appendix H - IPSEC

---

---

## Road Warriors

The prototypical "Road Warrior" is a traveller connecting to the Console Server from a laptop machine. For purposes of this document:

- Anyone with a dynamic IP address is a "Road Warrior."
- Any machine doing IPsec processing is a "gateway." Think of the single-user Road Warrior machine as a gateway with a degenerate subnet (one machine, itself) behind it.

These require somewhat different setup than VPN gateways with static addresses and with client systems behind them, but are basically not problematic. There are some difficulties which appear for some Road Warrior connections:

- Road Warriors who get their addresses via DHCP may have a problem. FreeS/WAN can quite happily build and use a tunnel to such an address, but when the DHCP lease expires, FreeS/WAN does not know that. The tunnel fails, and the only recovery method is to tear it down and rebuild it.
- If Network Address Translation (NAT) is applied between the two IPsec Gateways, this breaks IPsec. IPsec authenticates packets on an end-to-end basis, to ensure they are not altered en route. NAT rewrites packets as they go by.

In most situations, however, FreeS/WAN supports Road Warrior connections just fine.

## Configuration

### Before you start

#### Set up and test networking

Before trying to get FreeS/WAN working, you should configure and test IP networking on the Console Server and on the other end. IPsec cannot work without a working IP network beneath it.

Many reported "FreeS/WAN problems" turn out to actually be problems with routing or fire-walling. If any actual IPsec problems turn up, you often cannot even recognize them (much less debug them) unless the underlying network is right.



# Appendix H - IPSEC

---

## Enabling IPsec

The IPsec is disabled by default in the Console Server family. To enable it you must edit the file `/etc/inittab` and `/etc/config_files` and uncomment the lines regarding the IPsec. After performing these changes you must save the configuration using the `saveconf` tool and reboot the equipment.

## Quick Start

This is a quick guide to set up two common configurations: VPN and Road Warrior. There are three examples: one Road Warrior using RSA signature, one VPN using RSA signature and one VPN using shared secret(s). It will assume the other end is also running the FreeS/Wan. If it is not your case make the appropriate conversions for your IPsec software.

## "Road Warrior" remote access

A common requirement is for connections between a Console Server and some set of remote machines. For example, one administrator may want to access the Console Server from wherever he might be. We refer to the remote machines as "Road Warriors." For purposes of IPsec, anyone with a dynamic IP address is a Road Warrior.

### Information exchange

To set up a Road Warrior connection, you need some information about the system on the other end. Connection descriptions use *left* and *right* to designate the two ends. We adopt the convention that, from the Console Server's point of view, *left*=local and *right*=remote. The Console Server administrator needs to know some things about each Road Warrior:

- The system's public key (for RSA only).
- The ID that system uses in IPsec negotiation.

To get system's public key in a format suitable for insertion directly into the Console Server's `ipsec.conf` file, issue this command on the warrior machine:

```
/usr/local/sbin/ipsec showhostkey --right
```

The output should look like this (with the key shortened for easy reading):

```
rightrsasigkey=0s1LgR7/oUM...
```

The Road Warrior needs to know:

# Appendix H - IPSEC

---

---

- the Console Server's public key or the secret, and
- the ID the Console Server uses in IPsec negotiation

which can be generated by running `/usr/local/sbin/ipsec showhostkey -left` on the Console Server. Each warrior must also know:

- The IP address of the Console Server.

This information should be provided in a convenient format, ready for insertion in the warrior's `ipsec.conf` file. For example:

```
left=1.2.3.4
leftid=@gateway.example.com
leftrsasigkey=0s1LgR7/oUM...
```

The Console Server administrator typically needs to generate this only once. The same file can be given to all warriors.

Setup on the Road Warrior machine

Simply add a connection description *us-to-Console Server*, with the *left* and *right* information you gathered above to the `ipsec.conf` file. This might look like:

```
pre-configured link to Console Server
conn us-to-acs
 # information obtained from Console Server admin
 left=1.2.3.4 # Console Server IP address
 leftid=@acs.example.com
 # real keys are much longer than shown here
 leftrsasigkey=0s1LgR7/oUM...
 # our stuff
 right=%defaultroute
 rightid=@xy.example.com
```

# Appendix H - IPSEC

---

```
rightrsasigkey=0s1LgR7/oUM
```

Road warrior support on the Console Server

**Adding Road Warrior support so people can connect remotely to your Console Server is straightforward.**

```
conn gate-xy
```

```
 left=1.2.3.4
```

```
 leftid=@acs.example.com
```

```
 leftrsasigkey=0s1LgR7/oUM...
```

```
 # allow connection attempt from any address
```

```
 # attempt fails if caller cannot authenticate
```

```
 right=%any
```

```
 # authentication information
```

```
 rightid=@xy.example.com
```

```
 rightrsasigkey=0s1LgR7/oUM...
```

## ACS-to-network VPN

Often it may be useful to have explicitly configured IPsec tunnels between the Console Server and a gateway of an office with a fixed IP address (in this case every machine on the office network would have a secure connection with the Console Server), or between the Console Server and the Console Server administrator machine, which must, in this case, have a fixed IP address.

**To do it just insert this connection description to your ipsec.conf file with the variables that fit your environment:**

```
sample tunnel
```

```
The network here looks like:
```

```
ACS----acsnexthop.....righnexthop----right===rightsubnet
```

```
If ACS and right are on the same Ethernet, omit leftnexthop and
righnexthop.
```

# Appendix H - IPSEC

---

---

```
conn sample
 # ACS
 left=10.0.0.1
 leftid=@acs.example.com
 # next hop to reach right
 leftnexthop=10.44.55.66
 # This line is only for RSA signature
 leftrsasigkey=0s1LgR7/oUM...
 # right s.g., subnet behind it, and next hop to reach left
 right=10.12.12.1
 rightid=@xy.example.com
 rightnexthop=10.88.77.66
 rightsubnet=192.168.0.0/24
 auto=start
 # This line is only for RSA signature
 rightrrsasigkey=0s1LgR7/oUM...
 # This line is only for shared secret
 authby=secret
```

If you want to use shared secrets you must insert the following line to the `ipsec.secrets` file:

```
10.0.0.1 10.12.12.1 : PSK "secret"
```

The good part is that this connection descriptor and the secret line can be added to both the Console Server and the other end. This is the advantage of use `left` and `right` instead of using local remote parameters.

If you give an explicit IP address for *left* (and *left* and *right* are not directly connected), then you must specify *leftnexthop* (the router which *Console Server* sends packets to in order to

# Appendix H - IPSEC

---

get them delivered to *right*). Similarly, you may need to specify *rightnextop* (vice versa). The *nextop* parameters are needed because of an unfortunate interaction between FreeS/WAN and the Linuxkernel routing code. They will be eliminated in a future release, but perhaps not soon. We know they should go, but getting them out is not a simple problem.

## Setting up RSA authentication keys

To build a connection, the Console Server and the other end must be able to authenticate each other. For FreeS/WAN, the default is public key authentication based on the RSA algorithm. IPsec does allow several other authentication methods.

### Generating an RSA key pair

The Console Server doesn't have an RSA key pair by default. It will be generated on the first reboot after you have uncommented the IPsec lines in the file `/etc/rc.sysinit`. You also can generate your key pair by issuing the following commands as root:

- `/usr/local/sbin/ipsec newhostkey -bits <key length> -output /etc/ipsec.secrets`
- `chmod 600 /etc/ipsec.secrets`

Key generation may take some time. Also, it needs a lot of random numbers so it needs traffic on the Console Server ethernet port. The Console Server uses the traffic on the ethernet to generate its random numbers. It is also possible to use keys in other formats, not generated by FreeS/WAN. This may be necessary for interoperation with other IPsec implementations.

### Exchanging authentication keys

Once your gateway's key is in `ipsec.secrets`, the next step is to send your public key to everyone you need to set up connections with and collect their public keys. The other players will be:

- For a VPN: each gateway administrator needs public keys for all the other gateways his or her machine talks to.
- For a Road Warrior: the gateway needs public keys for all Warriors that connect to it, and each Warrior needs the gateway public key.

You need to extract the public part in a suitable format. This is done with the `ipsec_showhostkey` command. For VPN or Road Warrior applications, use one of the following:

# Appendix H - IPSEC

---

---

```
/usr/local/sbin/ipsec showhostkey --left
```

```
/usr/local/sbin/ipsec showhostkey --right
```

These two produce the key formatted for insertion in an `ipsec.conf` file.

Public keys need not be protected as fanatically as private keys. They are intended to be made public; the system is designed to work even if an enemy knows all the public keys used. You can safely make them publicly accessible. For example, put a gateway key on a Web page or make it available in DNS—or transmit it with an insecure method such as email.

## The Configuration File

### Description

The *ipsec.conf* file specifies most configuration and control information for the FreeS/WAN IPsec subsystem. (The major exception is secrets for authentication; `ipsec.secrets`) Its contents are not security-sensitive *unless* manual keying is being done for more than just testing, in which case the encryption/authentication keys in the descriptions for the manually-keyed connections are very sensitive (and those connection descriptions are probably best kept in a separate file, via the include facility described below).

The file is a text file, consisting of one or more *sections*. White space followed by # followed by anything to the end of the line is a comment and is ignored, as are empty lines which are not within a section.

A line which contains *include* and a file name, separated by white space, is replaced by the contents of that file, preceded and followed by empty lines. If the file name is not a full path-name, it is considered to be relative to the directory containing the including file. Such inclusions can be nested. Only a single filename may be supplied, and it may not contain white space, but it may include shell wildcards for example:

```
include ipsec.*.conf
```

The intention of the include facility is mostly to permit keeping information on connections, or sets of connections, separate from the main configuration file. This permits such connection descriptions to be changed, copied to the other security gateways involved, etc., without having to constantly extract them from the configuration file and then insert them back into

# Appendix H - IPSEC

---

it. Note the *also* parameter (described below) which permits splitting a single logical section (e.g., a connection description) into several actual sections.

A section begins with a line of the form:

*type name*

where *type* indicates what type of section follows, and *name* is an arbitrary name which distinguishes the section from others of the same type. (Names must start with a letter and may contain only letters, digits, periods, underscores, and hyphens.) All subsequent non-empty lines which begin with white space are part of the section; comments within a section must begin with white space too. There may be only one section of a given type with a given name.

Lines within the section are generally of the following form:

*parameter=value*

(Note the mandatory preceding white space.) There can be white space on either side of the =. Parameter names follow the same syntax as section names, and are specific to a section type. Unless otherwise explicitly specified, no parameter name may appear more than once in a section.

An empty *value* stands for the system default value (if any) of the parameter, i.e., it is roughly equivalent to omitting the parameter line entirely. A *value* may contain white space only if the entire *value* is enclosed in double quotes (""); a *value* cannot itself contain a double quote, nor may it be continued across more than one line.

Numeric values are specified to be either an integer (a sequence of digits) or a decimal number (sequence of digits optionally followed by "." and another sequence of digits).

There is currently one parameter which is available in any type of section:

*also*     The value is a section name; the parameters of that section are appended to this section, as if they had been written as part of it. The specified section must exist, must follow the current one, and must have the same section type. (Nesting is permitted, and there may be more than one also in a single section, although it is forbidden to append the same section more than once.) This allows, for example, keeping the encryption keys for a connection in a separate file from the rest of the description, by using both an also parameter and an *include* line.

# Appendix H - IPSEC

---

---

A section with name *%default* specifies defaults for sections of the same type. For each parameter in it, any section of that type which does not have a parameter of the same name gets a copy of the one from the *%default* section. There may be multiple *%default* sections of a given type, but only one default may be supplied for any specific parameter name, and all *%default* sections of a given type must precede all non-*%default* sections of that type. *%default* sections may not contain *also* parameters.

Currently there are two types of sections: a *config* section specifies general configuration information for IPsec, while a *conn* section specifies an IPsec connection.

## Conn Sections

A *conn* section contains a *connection specification*, defining a network connection to be made using IPsec. The name given is arbitrary, and is used to identify the connection to *ipsec\_auto* and *ipsec\_manual*. Here's a simple example:

```
conn snt
left=10.11.11.1
leftsubnet=10.0.1.0/24
leftnexthop=172.16.55.66
right=192.168.22.1
rightsubnet=10.0.2.0/24
rightnexthop=172.16.88.99
keyingtries=0 # be very persistent
```



**Terminology Note:** In automatic keying, there are two kinds of communications going on: transmission of user IP packets, and gateway-to-gateway negotiations for keying, rekeying, and general control. The data path (a set of “IPsec SAs”) used for user packets is hereon referred to as the “connection.” The path used for negotiations (built with “ISAKMP SAs”) is referred to as the “keying channel.”



# Appendix H - IPSEC

---

To avoid trivial editing of the configuration file to suit it to each system involved in a connection, connection specifications are written in terms of *left* and *right* participants, rather than in terms of local and remote. Which participant is considered *left* or *right* is arbitrary; IPsec figures out which one it is being run on based on internal information. This permits using identical connection specifications on both ends.

Many of the parameters relate to one participant or the other; only the ones for *left* are listed here, but every parameter whose name begins with *left* has a right counterpart, whose description is the same but with *left* and *right* reversed.

Parameters are optional unless marked *required*; a parameter required for manual keying need not be included for a connection which will use only automatic keying, and vice versa.

## Conn Parameters—General

The following parameters are relevant to both automatic and manual keying. Unless otherwise noted, for a connection to work, in general it is necessary for the two ends to agree exactly on the values of these parameters.

- type* The type of the connection. Currently the accepted values are: *tunnel* (the default) signifying a host-to-host, host-to-subnet, or subnet-to-subnet tunnel; *transport*, signifying host-to-host transport mode; and *passthrough* (supported only for manual keying), signifying that no IPsec processing should be done at all.
- left* Required. The IP address of the left participant's public-network interface. If it is the magic value *%defaultroute*, and *interfaces=%defaultroute* is used in the *config setup* section, *left* will be filled in automatically with the local address of the default-route interface (as determined at IPsec start-up time). This also overrides any value supplied for *leftnexthop*. (Either *left* or *right* may be *%defaultroute*, but not both.) The magic value *%any* signifies an address to be filled in (by automatic keying) during negotiation; the magic value *%opportunistic* signifies that both left and leftnexthop are to be filled in (by automatic keying) from DNS data for left's client.
- leftsubnet* Private subnet behind the left participant, expressed as *network/netmask*. If omitted, essentially assumed to be *left/32*, signifying that the left end of the connection goes to the left participant only.

# Appendix H - IPSEC

---

---

- leftnexthop* Next-hop gateway IP address for the left participant's connection to the public network. Defaults to %direct (meaning *right*).
- leftupdown* What *updown* script to run to adjust routing and/or firewalling when the status of the connection changes.

## Conn Parameters: Automatic Keying

The following parameters are relevant only to automatic keying, and are ignored in manual keying. Unless otherwise noted, for a connection to work, in general it is necessary for the two ends to agree exactly on the values of these parameters.

- auto* What operation, if any, should be done automatically at IPsec startup; currently-accepted values are *add* (signifying an *ipsec auto --add*), *route* (signifying that plus an *ipsec auto --route*), *start* (signifying that plus an *ipsec auto --up*), and *ignore* (also the default) (signifying no automatic startup operation). This parameter is ignored unless the *plutoload* or *plutostart* configuration parameter is set suitably; see the *config setup* discussion below.
- auth* Whether authentication should be done as part of ESP encryption, or separately using the AH protocol, acceptable values are *esp* (the default) and *ah*.
- authby* How the two security gateways should authenticate each other. Acceptable values are *secret* for shared secrets (the default) and *rsasig* for RSA digital signatures.
- leftid* How the left participant should be identified for authentication. Defaults to left. Can be an IP address or a fully-qualified domain name preceded by @ (which is used as a literal string and not resolved).

# Appendix H - IPSEC

---

- leftrsasigkey* The left participant's public key for RSA signature authentication, in RFC 2537 format. The magic value *%none* means the same as not specifying a value (useful to override a default). The value *%dnsondemand* means the key is to be fetched from DNS at the time it is needed. The value *%dnsonload* means the key is to be fetched from DNS at the time the connection description is read from *ipsec.conf*. Currently this is treated as *%none* if *right=%any* or *right=%opportunistic*. The value *%dns* is currently treated as *%dnsonload* but will change to *%dnsondemand* in the future. The identity used for the left participant must be a specific host, not *%any* or another magic value. *Caution:* if two connection descriptions specify different public keys for the same *leftid*, confusion and madness will ensue.
- pfs* Whether Perfect Forward Secrecy of keys is desired on the connection's keying channel. (With PFS, penetration of the key-exchange protocol does not compromise keys negotiated earlier.) Acceptable values are *yes* (the default) and *no*.
- keylife* How long a particular instance of a connection (a set of encryption/authentication keys for user packets) should last, from successful negotiation to expiry; acceptable values are an integer optionally followed by *s* (a time in seconds) or a decimal number followed by *m*, *h*, or *d* (a time in minutes, hours, or days respectively) (default *8.0h*, maximum *24h*).
- rekey* Whether a connection should be renegotiated when it is about to expire; acceptable values are *yes* (the default) and *no*.
- rekeymargin* How long before connection expiry or keying-channel expiry should attempts to negotiate a replacement begin; acceptable values as for *keylife* (default *9m*).
- rekeyfuzz* Maximum percentage by which *rekeymargin* should be randomly increased to randomize rekeying intervals (important for hosts with many connections). Acceptable values are an integer, which may exceed 100, followed by a "%."
- keyingtries* how many attempts (an integer) should be made to negotiate a connection, or a replacement for one, before giving up (default *3*). The value *0* means "never give up."
- ikelifetime* How long the keying channel of a connection (buzzphrase: ISAKMP SA) should last before being renegotiated. Acceptable values as for *keylife*.

# Appendix H - IPSEC

---

---

*compress* Whether IPComp compression of content is desired on the connection. Acceptable values are *yes* and *no* (the default).

## Conn Parameters: Manual Keying

The following parameters are relevant only to manual keying, and are ignored in automatic keying. Unless otherwise noted, for a connection to work, in general it is necessary for the two ends to agree exactly on the values of these parameters. A manually-keyed connection must specify at least one of AH or ESP.

*spi or spibase* This or *spibase* required for manual keying) the SPI number to be used for the connection. Must be of the form *0xhex*, where *hex* is one or more hexadecimal digits. (Note: it will generally be necessary to make *spi* at least *0x100* to be acceptable to KLIPS, and use of SPIs in the range *0x100-0xfff* is recommended.)

*esp* ESP encryption/authentication algorithm to be used for the connection, e.g. *3des-md5-96*.

*espenckey* ESP encryption key.

*espauthkey* ESP authentication key.

*espreplay\_window* ESP replay-window setting. An integer from 0 to 64. Relevant only if ESP authentication is being used.

*leftespsi* SPI to be used for the leftward ESP SA, overriding automatic assignment using *spi* or *spibase*. Typically a hexadecimal number beginning with 0x.

*ah* AH authentication algorithm to be used for the connection, e.g. *hmac-md5-96*. Default is not to use AH.

*ahkey* Required if *ah* is present. AH authentication key

*ahreplay\_window* AH replay-window setting. An integer from 0 to 64.

*leftahspi* SPI to be used for the leftward AH SA, overriding automatic assignment using *spi* or *spibase*. Typically a hexadecimal number beginning with 0x.

# Appendix H - IPSEC

---

## Config Sections

At present, the only config section known to the IPsec software is the one named `setup`, which contains information used when the software is being started. Here's an example:

```
config setup
 interfaces="ipsec0=eth1 ipsec1=ppp0"
 klipsdebug=none
 plutodebug=all
 manualstart=
 plutoload="snta sntb sntc sntd"
 plutostart=
```

Parameters are optional unless marked “(required).” The currently-accepted *parameter* names in a *config setup* section are:

|                       |                                                                                                                                                                                                                                                                                                            |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>interfaces</i>     | Required. Virtual and physical interfaces for IPsec to use: a single <i>virtual=physical</i> pair, a (quoted!) list of pairs separated by white space, or %defaultroute, which means to find the interface <i>d</i> that the default route points to, and then act as if the value was “ipsec0= <i>d</i> ” |
| <i>forwardcontrol</i> | Whether <i>setup</i> should turn IP forwarding on (if it's not already on) as IPsec is started, and turn it off again (if it was off) as IPsec is stopped. Acceptable values are yes and (the default) no.                                                                                                 |
| <i>klipsdebug</i>     | How much KLIPS debugging output should be logged. An empty value, or the magic value <i>none</i> , means no debugging output (the default). The magic value <i>all</i> means full output.                                                                                                                  |
| <i>plutodebug</i>     | How much Pluto debugging output should be logged. An empty value, or the magic value <i>none</i> , means no debugging output (the default). The magic value <i>all</i> means full output.                                                                                                                  |
| <i>dumpdir</i>        | In what directory should things started by <i>setup</i> (notably the Pluto daemon) be allowed to dump core? The empty value (the default) means they are not allowed to.                                                                                                                                   |

# Appendix H - IPSEC

---

---

|                      |                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>manualstart</i>   | Which manually-keyed connections to set up at startup (empty, a name, or a quoted list of names separated by white space).                                                                                                                                                                                                                                                                          |
| <i>plutoload</i>     | Which connections (by name) to load into Pluto's internal database at startup (empty, a name, or a quoted list of names separated by white space); see <i>ipsec_auto</i> for details. Default is none. If the special value %search is used, all connections with auto=add, auto=route, or auto=start are loaded.                                                                                   |
| <i>plutostart</i>    | Which connections (by name) to attempt to negotiate at startup (empty, a name, or a quoted list of names separated by white space). Any such names which do not appear in <i>plutoload</i> are implicitly added to it. Default is none. If the special value %search is used, all connections with auto=route or auto=start are routed, and all connections with auto=start are started.            |
| <i>plutowait</i>     | Should Pluto wait for each <i>plutostart</i> negotiation attempt to finish before proceeding with the next? Values are yes (the default) or no.                                                                                                                                                                                                                                                     |
| <i>prepluto</i>      | Shell command to run before starting Pluto. For example, to decrypt an encrypted copy of the <i>ipsec.secrets</i> file). It's run in a very simple way. Complexities like I/O redirection are best hidden within a script. Any output is redirected for logging, so running interactive commands is difficult unless they use <i>/dev/tty</i> or equivalent for their interaction. Default is none. |
| <i>postpluto</i>     | Shell command to run after starting Pluto (e.g., to remove a decrypted copy of the <i>ipsec.secrets</i> file).                                                                                                                                                                                                                                                                                      |
| <i>fragicmp</i>      | Whether a tunnel's need to fragment a packet should be reported back with an ICMP message, in an attempt to make the sender lower his PMTU estimate. Acceptable values are yes (the default) and no.                                                                                                                                                                                                |
| <i>packetdefault</i> | What should be done with a packet which reaches KLIPS (via a route into a virtual interface) but does not match any route;. Acceptable values are pass ( <i>insecure unless you really know what you're doing!!!</i> ), drop (the default), and reject (currently same as drop).                                                                                                                    |
| <i>hidetos</i>       | Whether a tunnel packet's TOS field should be set to 0 rather than copied from the user packet inside. Acceptable values are yes (the default) and no.                                                                                                                                                                                                                                              |

# Appendix H - IPSEC

---

- uniqueids* Whether a particular participant ID should be kept unique, with any new (automatically keyed) connection using an ID from a different IP address deemed to replace all old ones using that ID. Acceptable values are yes and no (the default).
- overridemtu* Value that the MTU of the ipsec*n* interface(s) should be set to, overriding IPsec's (large) default. This parameter is needed only in special situations.

## Recommended Configuration

Certain parameters are now strongly-recommended defaults, but cannot (yet) be made system defaults due to backward compatibility. Recommended config setup parameters are:

- `plutoload=%search`
- `plutostart=%search`

In practice, it is preferable to use the auto parameter to control whether a particular connection is added or started automatically.

- uniqueids=yes* Participant IDs normally *are* unique, so a new (automatically-keyed) connection using the same ID is almost invariably intended to replace an old one.

Recommended *conn* parameters (mostly for automatic keying, as manual keying seldom sees much use) are:

- keyingtries=0* Unlimited retries are normally appropriate for VPN connections. Finite values may be needed for Road Warrior and other more ephemeral applications, but the fixed small default is pretty much useless.
- disablearrivalcheck=no* Tunnel-exit checks improve security and do not break any normal configuration.
- authby=rsasig* Digital signatures are superior in every way to shared secrets.

# Appendix H - IPSEC

---

---

## IPsec Usage

This section will teach you:

- How to start and stop the IPsec daemon.
- How to add and remove an IPsec connection from the IPsec database.
- How to start and stop a connection.

### The IPsec Daemon

The ipsec daemon is automatically initialized when you boot your Console Server equipment. However if you want not to use the IPsec you may want to disable the auto run. To make it just comment the lines regarding the IPsec on the /etc/rc.sysinit script. You can also use the command:

```
/usr/local/sbin/ipsec setup
```

to start and stop the IPsec daemon. This program accept the options: --start, --stop and restart.

### Adding and Removing a Connection

All the connections can be loaded to the IPsec database at boot time if these connections have the auto parameter set to add. However if a certain connection doesn't have this option set and you wish to add this connection manually you can use the following command:

```
/usr/local/sbin/ipsec (auto/manual) --add <connection name>
```

You must use auto or manual depending on your connection keying type (manual/auto). Similarly to take a connection out of the IPsec database you can use the command:

```
/usr/local/sbin/ipsec (auto/manual) --delete <connection name>
```

Once a connection descriptor is in the IPsec internal database, IPsec will accept the other end to start the security connection negotiation. You can also start its negotiation as explained in the next chapter.



# Appendix H - IPSEC

---

## Starting and Stopping a Connection

All the connections can be negotiated at boot time if these connections have the `auto` parameter set to `start`. However if a certain connection doesn't have this option set you can set it. Once a connection descriptor is in the IPsec internal database, you can start its negotiation using the command:

```
/usr/local/sbin/ipsec (auto/manual) --up <connection name>
```

Similarly to close a tunnel you use the command:

```
/usr/local/sbin/ipsec (auto/manual) --down <connection name>
```

# List of Wiz Application Parameters

---

---

## Basic Parameters (wiz)

- Hostname
- System IP
- Domain Name
- DNS Server
- Gateway IP
- Network Mask

## Authentication Parameters (wiz --auth)

- Authtype
- Authhost1
- Accthost1
- Authhost2
- Accthost2
- Radtimeout
- Radretries
- Secret

# List of Wiz Application Parameters

---

---

## Terminal Appearance Parameters (`wiz --tl`)

- Issue
- Prompt

## Alarm Parameter (`wiz --al`)

- Alarm

## Data Buffering Parameters (`wiz --db`)

- Data\_buffering
- Conf.nfs\_data\_buffering
- Syslog\_buffering
- Dont\_show\_DBmenu
- DB\_timestamp
- DB\_mode

# List of Wiz Application Parameters

---

---

## Sniffing Parameters (wiz --snf)

- Admin\_users
- Sniff\_mode
- Escape\_char
- Multiple\_sessions

## Syslog Parameters (wiz --sl)

- Conf.facility
- Conf.DB\_facility

## Terminal Server Profile Other Parameters (wiz --tso)

- Host
- Term
- Conf.locallogins

# List of Wiz Application Parameters

---

---

## Access Method Parameters (wiz --ac <type>)

### (CAS profile)

- Ipno
- Socket\_port
- Protocol
- Users
- Poll\_interval
- Tx\_interval
- Idletimeout
- Conf.group
- <sN>.serverfarm

### (TS profile)

- Protocol
- Socket\_port
- Userauto

# List of Wiz Application Parameters

---

## Serial Settings Parameters (`wiz --sset <type>`)

(CAS profile)

- Speed
- Datasize
- Stopbits
- Parity
- Flow
- Dcd
- SttyCmd
- DTR\_reset

(TS profile)

- Speed
- Datasize
- Stopbits
- Parity
- Flow
- Dcd

# List of Figures

---

|                                                                                      |     |
|--------------------------------------------------------------------------------------|-----|
| 1. Console Access Server diagram . . . . .                                           | 18  |
| 2. CAS diagram with various authentication methods . . . . .                         | 19  |
| 3. The AlterPath ACS32, its cables, connectors and other box contents . . . . .      | 20  |
| 4. The AlterPath ACS16, its cables, connectors and other box contents . . . . .      | 21  |
| 5. The initial wizard configuration screen . . . . .                                 | 31  |
| 6. Login page of Web Configuration Manager . . . . .                                 | 34  |
| 7. Configuration & Administration Menu page . . . . .                                | 34  |
| 8. General page . . . . .                                                            | 35  |
| 9. The initial wizard configuration screen . . . . .                                 | 41  |
| 10. Choose a free COM port . . . . .                                                 | 45  |
| 11. Port Settings . . . . .                                                          | 46  |
| 12. The /etc/hostname file with hostname typed in . . . . .                          | 48  |
| 13. Contents of the /etc/hosts file . . . . .                                        | 48  |
| 14. Configuration and Administration page . . . . .                                  | 64  |
| 15. Port Selection page . . . . .                                                    | 65  |
| 16. Serial Port Configuration page - top . . . . .                                   | 65  |
| 17. Profile Section of Serial Port Configuration page. . . . .                       | 66  |
| 18. Serial Ports - Users Group Table Entry page . . . . .                            | 67  |
| 19. An example using the Clustering feature . . . . .                                | 94  |
| 20. Example of Centralized Management . . . . .                                      | 99  |
| 21. Edit Text File page . . . . .                                                    | 104 |
| 22. Data Buffering section of the Serial Port Configuration page - menu dropdown . . | 109 |
| 23. Data Buffering section of the Serial Port Configuration page - mode dropdown . . | 110 |
| 24. Data Buffering section of the General page . . . . .                             | 110 |

# List of Figures

---

---

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| 25. Page 1 of IPTables filtering . . . . .                                | 124 |
| 26. IPTables filter Information page . . . . .                            | 125 |
| 27. IP Statistics page . . . . .                                          | 126 |
| 28. Ports configured for Dial-in Access . . . . .                         | 151 |
| 29. Terminal Server diagram . . . . .                                     | 153 |
| 30. Sniff Session section of the Serial Port Configuration page . . . . . | 175 |
| 31. Syslog page 1 . . . . .                                               | 184 |
| 32. Cable 1 - Cyclades RJ-45 to DB-25 Male, Straight Through . . . . .    | 229 |
| 33. Cable 2 - Cyclades RJ-45 to DB-25 Female/Male, Crossover . . . . .    | 230 |
| 34. Cable 3 - Cyclades RJ-45 to DB-9 Female, Crossover . . . . .          | 230 |
| 35. Cable 4 - Cyclades RJ-45 to Cyclades RJ-45, Crossover . . . . .       | 231 |
| 36. Loop-Back Connector . . . . .                                         | 231 |
| 37. CAT.5e Inline Coupler./Sun Netra Adapter . . . . .                    | 232 |
| 38. RJ-45 Female to DB-25 Male Adapter . . . . .                          | 232 |
| 39. RJ-45 Female to DB-25 Female Adapter . . . . .                        | 233 |
| 40. RJ-45 Female to DB-9 Male Adapter . . . . .                           | 233 |
| 41. RJ-45 Female to DB-9 Female Adapter . . . . .                         | 234 |
| 42. DB-25 Male to DB-9 Female Adapter . . . . .                           | 234 |
| 43. Data flow diagram of Linux-PAM . . . . .                              | 256 |
| 44. Initial test . . . . .                                                | 281 |
| 45. Second screen, showing changed positions . . . . .                    | 282 |
| 46. Serial Port Connection page . . . . .                                 | 288 |
| 47. Port Connection page . . . . .                                        | 289 |
| 48. SSH User Authentication page . . . . .                                | 289 |



# List of Tables

---

|                                                                                                      |     |
|------------------------------------------------------------------------------------------------------|-----|
| 1. Hardware vs. Configuration Methods .....                                                          | 27  |
| 2. Configuration Section .....                                                                       | 37  |
| 3. Web User Management Section .....                                                                 | 38  |
| 4. Administration Section .....                                                                      | 38  |
| 5. Information Section .....                                                                         | 39  |
| 6. Master Cyclades Configuration (where it differs from the CAS standard) .....                      | 95  |
| 7. AlterPath Console Server configuration for Slave-1 (where it differs from the CAS standard) ..... | 97  |
| 8. AlterPath Console Server configuration for Slave-2 (where it differs from the CAS standard) ..... | 97  |
| 9. General Options for the Help Wizard .....                                                         | 140 |
| 10. Help CLI Options - Synopsis 1 .....                                                              | 141 |
| 11. Help CLI Options - Synopsis 2 .....                                                              | 142 |
| 12. vi modes .....                                                                                   | 212 |
| 13. vi navigation commands .....                                                                     | 213 |
| 14. vi file modification commands .....                                                              | 213 |
| 15. vi line mode commands .....                                                                      | 214 |
| 16. Process table .....                                                                              | 219 |
| 17. AlterPath Console Server power requirements .....                                                | 223 |
| 18. AlterPath Console Server environmental conditions .....                                          | 223 |
| 19. AlterPath CS physical specifications .....                                                       | 224 |
| 20. AlterPath Console Server safety specifications .....                                             | 224 |
| 21. Cables and their pin specifications .....                                                        | 227 |
| 22. Which cable to use .....                                                                         | 228 |
| 23. Parameters Common to CAS, TS, & Dial-in Access .....                                             | 235 |

# List of Tables

---

---

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| 24. Mostly CAS-specific Parameters .....                                  | 239 |
| 25. TS Parameters .....                                                   | 251 |
| 26. Dial-in configuration Parameters .....                                | 252 |
| 27. Files to be included in /etc/config_file and the program to use ..... | 278 |
| 28. Windows XP + JREv1.4.0_01 or 02 .....                                 | 286 |
| 29. Redhat 7.3 + JREv1.4.0_01 or 02 .....                                 | 286 |
| 30. CPU LED Code Interpretation .....                                     | 291 |
| 31. Required information for the OpenSSL package .....                    | 292 |

# Glossary

---

## Authentication

Authentication is the process of identifying an individual, usually based on a username and password. In security systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their identity. Authentication merely ensures that the individual is who he or she claims to be, but says nothing about the access rights of the individual. (Source: Webopedia)

## Break Signal

A break signal is generated in an RS-232 serial line by keeping the line in zero for longer than a character time. Breaks at a serial console port are interpreted by Sun servers as a signal to suspend operation and switch to monitor mode.

## Console Access Server (CAS)

A CAS has an Ethernet LAN connection and many RS-232 serial ports. It connects to the console ports of servers and networking equipment and allows convenient and secure access from a single location.

## Console Port

Most of the equipment in a data center (servers, routers, switches, UPS, PBX, etc.) has a serial console port for out-of-band management purposes.

## Cluster

A cluster is a group of one or more computers working as a group to execute a certain task. From the user standpoint, a cluster acts as a large computer system.

## Flash

Flash refers to a type of memory that can be erased and reprogrammed in units of memory known as blocks rather than one byte at a time; thus, making updating to memory easier.

## In-band network management

In a computer network, when the management data is accessed using the same network that carries the data, this is called “in-band management.”

# Glossary

---

---

## IP packet filtering

This is a set of facilities in network equipment that allows the filtering of data packets based on source/destination addresses, protocol, TCP port number and other parameters. Packet filtering is one of the main functions of a firewall.

## KVM Switch (KVM)

Keyboard-Video-Mouse Switches connect to the KVM ports of many computers and allow the network manager to access them from a single KVM station.

## Mainframe

Large, monolithic computer system.

## MIBs

Management Information Bases. SNMP-compliant devices, called agents, store data about themselves in MIBs and return this data to the SNMP requesters.

## Out-of-band network management

In a computer network, when the management data is accessed through a network that is independent of the network used to carry data, this is called “out-of-band network management.”

## Off-line data buffering

This is a CAS feature that allows capture of console data even when there is no one connected to the port.

## Profile

Usage setup of the AlterPath Console Server: either as a Console Access Server (CAS), a Terminal Server, or a Remote Access Server.

## RADIUS

Protocol between an authentication server and an access server to authenticate users trying to connect to the network.

# Glossary

---

## RISC

Reduced Instruction Set Computer. This describes a computer processor architecture that uses a reduced set of instructions (and achieves performance by executing those instructions very fast.) Most UNIX servers (Sun Sparc, HP, IBM RS6000, Compaq Alpha) were designed with a processor using a RISC architecture. The Intel<sup>®</sup> x86 architecture.

## RS-232

A set of standards for serial communication between electronic equipment defined by the Electronic Industries Association in 1969. Today, RS-232 is still widely used for low-speed data communication.

## Secure Shell (SSH)

SSH has the same functionality as Telnet (see definition below), but adds security by encrypting data before sending it through the network.

## Server Farm

A collection of servers running in the same location (see Cluster).

## SNMP

Short for Simple Network Management Protocol, a set of protocols for managing complex networks. The first versions of SNMP were developed in the early 80s. SNMP works by sending messages, called protocol data units (PDUs), to different parts of a network. SNMP-compliant devices, called agents, store data about themselves in Management Information Bases (MIBs) and return this data to the SNMP requesters. (Source: Webopedia)

## Telnet

Telnet is the standard set of protocols for terminal emulation between computers over a TCP/IP connection. It is a terminal emulation program for TCP/IP networks such as the Internet. The Telnet program runs on your computer and connects your PC to a server on the network. You can then enter commands through the Telnet program and they will be executed as if you were entering them directly on the server console. This enables you to control the server and communicate with other servers on the network. To start a Telnet session, you must log in to a server by entering a valid username and password. Telnet is a common way to remotely control Web servers. (from webopedia.com)

# Glossary

---

---

## Terminal Server

A terminal server has one Ethernet LAN port and many RS-232 serial ports. It is used to connect many terminals to the network. Because they have the same physical interfaces, terminal servers are sometimes used as console access servers.

## TTY

The UNIX name for the COM (Microsoft) port.

## U Rack height unit

A standard computer rack has an internal width of 17 inches. Rack space on a standard rack is measured in units of height (U). One U is 1.75 inches. A device that has a height of 3.5 inches takes 2U of rack space.

# Index

---

## A

Access Method 61  
Alarm 133  
Authentication 84

## B

Basic Wizard 56  
Battery 23

## C

Cable Length 226  
CAS Setup Scenario 249  
CLI 27  
Clustering 93  
Command Line Interface 27, 55  
Configuration using a Web browser 33  
Connectors 226  
CronD 102  
Custom Wizard 29  
Customization Process 274

## D

Data Buffers 105  
Default Configuration Parameters 27  
DHCP 116  
DNS Server 29  
Domain 29  
Dual Power Management 119

## E

Ethernet 28

## F

Filters 120  
Flash Memory Loss 277

## G

Gateway 28  
    default 29  
Generating Alarms 126

## H

Hardware Specifications 223  
Hardware Test 280  
HyperTerminal 28

## I

init process 273  
IP Address 29  
IPsec 295

## K

Kerberos 52, 85, 88, 263  
Kermit 28

## L

LDAP 263  
Linux File Structure 210  
Linux-PAM 255  
loop-back connector 20

# Index

---

## M

Minicom 28

## N

Netmask 29

NTP 143

## P

Passwords 210

PCMCIA 146

Port Test 280

pslave.conf file 235

## R

Radius authentication 151

RJ-45 20

Routing Table 214

RS-232 Standard 224

## S

Secure Shell Session 215

Sendmail 133

Sendsms 133

serial ports 20

SNMP 180

Snmpttrap 133

Sun Netra Crossover cable 20

Syslog-n 188

System Requirements 26

## T

Telnet 37

Terminal Appearance 202

Time Zone 207

## U

Upgrades 275

Using 60

Using the Wizard through your Browser 60

## W

Wizard 29