



Sun Fire™ B10n Content Load Balancing Blade Version 1.3 Administration Guide

Sun Microsystems, Inc.
www.sun.com

Part No. 817-6210-10
June 2004, Revision A

Submit comments about this document at: <http://www.sun.com/hwdocs/feedback>

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook2, docs.sun.com, Sun Fire, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and in other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in the Sun Microsystems, Inc. license agreements and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct. 1998), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook2, docs.sun.com, Sun Fire, et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

LA DOCUMENTATION EST FOURNIE "EN L'ÉTAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.



Contents

1. Product Overview	1
Hardware and Software Overview	1
Software Architecture	2
Supported Hardware and Software	3
Product Features	6
Key Features	6
Server Selection Algorithms	7
Supported Protocols	7
Diagnostic Support	7
Weighted Least Connections Feature	8
Testing Weighted Least Connections	9
Application Monitoring Feature	9
The Role of the Content Load Balancing Blade	9
Load Balancing	12
Topology Fundamentals	14
The Role of the SSL Proxy Blade	16
Failover Alternatives	16
The Role of VLANs	17
System Integration	19

Usage Overview 19

Command Modes 20

2. Installing the Blade and Setting Up the System 21

Installing the Hardware 21

▼ To Install the Hardware 21

LED Displays 25

Location of Ports 26

Connecting to the 10/100/1000BASE-T Data Network Ports 28

Serial Port Pin Numbers 29

Powering On Content Load Balancing Blades 30

Powering Off Content Load Balancing Blades 32

Powering Off With an Orderly Shutdown of the Operating System 32

Forcing the Power Off 32

Powering Off a Load Balancing Blade Without Requiring the Confirmation Prompt 33

Powering a Load Balancing Blade Down to Standby Mode 33

Powering Off a Content Load Balancing Blade to Remove It 34

Replacing Your Sun Fire B10n Blade 35

▼ To Export the Configuration From the Old Board 35

▼ To Import the Configuration to the Upgraded Board 36

3. Preparing the Sun Fire B10n Blade for Load Balancing 37

Configuring the Blade Servers 37

▼ To Check the Blade Server Module Software Version 38

▼ To Set Up a Solaris SPARC Blade Server Module 38

Setting Up the Load Balancing Blade 41

▼ To Set Up a Content Load Balancing Blade 41

Completing the Basic Configuration 44

▼ To Configure a Default Gateway 44

- ▼ To Configure the DNS Server 44
- ▼ To Configure DNS Suffix 44
- ▼ To Commit the Configuration 44
- ▼ To Verify the Configuration 45

4. Command-Line Options 47

Typographic Conventions Specific to the Sun Fire B10n Command Line Interface 48

User Access 48

Using the `login` Command 49

- ▼ To Log In as Administrator 49

Adding Users 50

- ▼ To Add a User 50
- ▼ To Change the User Access Level 51
- ▼ To Change the User Password 51
- ▼ To Remove a User 51
- ▼ To List All Users 51

Configuring the Networking 52

- ▼ To Configure the Management IP Address 52
- ▼ To Send a `ping` Request 53
- ▼ To Unconfigure a Network Interface 54
- ▼ To Configure a DNS Server 54
- ▼ To Remove a DNS Server 54
- ▼ To Configure the DNS Suffix 55
- ▼ To Show DNS Use 55
- ▼ To Unconfigure the DNS Suffix 55
- ▼ To Configure the Default Gateway 55
- ▼ To Unconfigure the Default Gateway 56
- ▼ To Set the Default Hostname 56

- ▼ To Show the Network Configurations 56
- ▼ To Show ARP Entries 57
- ▼ To List the VIPs Configured 57
- ▼ To Show All End Points 58

Monitoring 58

- ▼ To Set Up Server Monitoring 58
- ▼ To Set Up Application Monitoring 59
- ▼ To Configure Application Monitoring Parameters 60
- ▼ Enable or Disable Application Monitoring for a Specified Service 61

Configuring SSL Device Entries 61

- ▼ To Add an SSL Device 61
- ▼ To Remove an SSL Device 62
- ▼ To Add a Port Pair to an SSL Device Entry 63
- ▼ To Remove a Port Pair from an SSL Device Entry 64
- ▼ To Add an Interface to an SSL Device Entry 65
- ▼ To Remove an Interface from an SSL Device Entry 65
- ▼ To Enable an SSL Device Entry 66
- ▼ To Disable an SSL Device Entry 66
- ▼ To Show the Configured SSL Devices 67

Configuring Multiple SSL Devices 67

Configuring the Content Load Balancing Blade 67

- ▼ To Set the TCP Parameters 68
 - Defaults 68
- ▼ To Set Parameters for TCP Connection Handoff 69
- ▼ To Show All the Default TCP Parameters Settings 70

Load Balancing Service Configuration and Management 70

Creating a Load Balancing Service 70

- ▼ To Create a Load Balancing Service 71

Defaults	72
▼ To Configure the Subnet Mask for a VIP	73
▼ To Add SSL Devices to a Service	73
Usage Guidelines	75
▼ To Remove SSL Devices From a Service	75
▼ To Set SSL Devices in a Service as Active or Standby	76
▼ To Add a Default Load Balancing Group to a Load Balancing Service	76
Defaults	78
▼ To Set the TCP Parameters for a Service	78
▼ To Set Parameters for TCP Connection Handoff for a Service	80
Defaults	80
▼ To Add a Service Point to a Service	81
▼ To Remove a Service Point From a Service	82
▼ To Configure a Service for Client IP or Subnet-Based Persistence	83
Defaults	84
▼ To Remove Client IP or Subnet Based Persistence from a Service	85
Configuring a Service for Service Point Tracking	85
▼ To Configure a Service for Service Point Tracking	86
Defaults	86
▼ To Remove Tracking Service Point from a Service	87
▼ To Configure a Service for Cookie-Based Persistence	88
▼ To Remove Cookie Persistence From a Service	89
▼ To Enable a Load Balancing Service	90
▼ To Disable a Load Balancing Service	90
▼ To Remove a Load Balancing Service	91
Server Configuration	91
▼ To Enable a Server	91
▼ To Disable a Server	92

▼ To Remove a Server From All Services	93
Load Balancing Rule Configuration	93
Creating an IP Load Balancing Rule	93
▼ To Create an IP Load Balancing Rule	94
Creating an HTTP Load Balancing Rule	95
▼ To Create an HTTP Load Balancing Rule	95
▼ To Remove a Load Balancing Rule	96
▼ To Build Load Balancing Rules	97
▼ To Get the Status for the Build for Load Balancing Rules	97
▼ To Get the Status for the Last Build for Load Balancing Rules	98
▼ To Stop the Build for Load Balancing Rules	98
Load Balancing Group Configuration and Management	98
▼ To Create a Default Load Balancing Group	99
▼ To Create a Load Balancing Group	99
▼ To Add Rules to a Load Balancing Group	101
▼ To Remove Rules From a Load Balancing Group	103
▼ To Add Servers to a Load Balancing Group	103
▼ To Remove Servers From a Load Balancing Group	104
▼ To Set Servers for a Load Balancing Group as Active or Standby	105
▼ To Modify the Load Balancing Scheme of a Load Balancing Group	107
▼ To Modify the Weight of a Server in a Load Balancing Group	107
▼ To Remove Load Balancing Groups	108
Load Balancing Configuration Listings	109
▼ To List Load Balancing Services	109
▼ To List Load Balancing Rules	110
▼ To List Load Balancing Groups	110
▼ To List Servers	111
Configuring the System	112

- ▼ To Configure the Image for the Next Reboot 112
- ▼ To Download a New Boot Image Over the Network 113
- ▼ To Configure the Diagnostics Level 114
- ▼ To Configure the Debug Level for Specific Modules 114
- ▼ To Shutdown the System 116
- ▼ To Reboot the System 116
- ▼ To Show the Date and Time 117
- ▼ To Show the System Settings on the B10n 117
- ▼ To Show the System Uptime 117
- ▼ To Show All of the Blade Configurations on the B10n 118
- ▼ To Compare the Running Configuration With the Saved Configuration 118
- ▼ To Show the Configuration in Running Memory 118
- ▼ To Show the Configuration Saved in Flash Memory 119
- ▼ To Show Memory 119
- ▼ To Show Modules 120
- ▼ To Export a File to a Remote Host 124
- ▼ To Import a File From a Remote Host 125
- ▼ To Commit the Current Configuration 125
- ▼ To Show the Current Configuration in Flash 125
- ▼ To Save this Current Configuration in Flash 126
- ▼ To Remove the Current Configuration in Flash 126
- ▼ To Specify the Configuration in Flash to Use After a System Reboot 126
- Flash File System Commands 127
 - ▼ To Check or Repair the Flash File System 127
 - ▼ To Output a File to the Screen 127
 - ▼ To Change the Current Directory 128
 - ▼ To Copy a File 128
 - ▼ To Rename a File 128

- ▼ To Delete a File 128
- ▼ To Create a New Directory 128
- ▼ To Remove a Directory 129
- ▼ To List Files 129
- ▼ To Print the Current Working Directory 129
- ▼ To Compress All Files in a Directory 129
- ▼ To Uncompress Files 130
- ▼ To Display Contents of a Compressed File 130

Other Useful Commands 130

- ▼ To Clear the Screen 130
- ▼ To Create an Alias for Any Command 130
- ▼ To Send a Message to All Logged-on Users 131
- ▼ To Echo a String on the Screen 131
- ▼ To View the Command-Line Interface Tree 131
- ▼ To Print the History of All Executed Commands 138
- ▼ To Get Help for CLI Commands 139
- ▼ To Logout 139
- ▼ To Exit From a Script 139
- ▼ To Retrieve the Current User Information 139
- ▼ To Retrieve Information About All Users 140
- ▼ To Display Console Settings 140
- ▼ To Turn On Hardwrap on the Console 140

5. Configuring Failover 141

Configuring Path Failover 142

- ▼ To Configure IPMP On a Sun Fire B100 When Using Interfaces on a Sun Fire B100 as the Target Paths 142
- ▼ To Add a Path Failover Target Address to an Interface 144
- ▼ To Remove a Path Failover Target Address on an Interface 144

- ▼ To Enable Path Failover Monitoring 145
- ▼ To Disable Path Failover Monitoring 145
- ▼ To Configure Path Failover Monitoring Parameters 145
- ▼ To Show the Path Failover Status 146
- Sample Configuration 147
- Path-Failover-Monitor 147

Configuring Blade Failover 148

- ▼ To Configure the Failover Peer IP Addresses 149
- ▼ To Enable Failover Monitoring 149
- ▼ To Disable Failover Monitoring 150
- ▼ To Start Failover 150
- ▼ To Skip the Failover Synchronization at Boot Time 151
- ▼ To Stop the Failover Synchronization 151
- ▼ To Configure Failover Monitoring 152
- ▼ To Force Failover 152
- ▼ To Sync the Failover Configuration 153
- ▼ To Remove the Failover State File 153
- ▼ To Remove the Current Configuration 154
- ▼ To Remove the Failover Configuration 154
- ▼ To List the Failover Configurations 155
- List of Configuration Commands 156

6. Configuring VLAN Parameters 159

Available VLAN Types 159

Enabling and Disabling VLAN Tagging 160

- ▼ To Enable VLAN Tagging 160
- ▼ To Set Client VLAN Tagging 161
- ▼ To Set Management VLAN Tagging 161
- ▼ To Disable VLAN Tagging 162

▼	To Enable VLAN Tagging for a Service	163
▼	To Set VLAN for Service	163
▼	To Disable VLAN Tagging for a Service	164
▼	To Show VLANs	165
7.	Updating the Application Software and the BSC Firmware	167
	Introduction	167
	Setting up a TFTP Server	168
	Software Architecture	169
	Upgrading and Downgrading Software on the Sun Fire B10n Load Balancer	170
	To Upgrade to Version 1.1, 1.2, or 1.3.5 From Version 1.0	170
	To Upgrade to Version 1.2 or 1.3.5 From Version 1.1	171
	To Downgrade to Version 1.0 From Version 1.1 to 1.3.5	171
	To Downgrade to Version 1.1 From Version 1.2 to 1.3.5	171
	Updating B10n Application Software and BSC Firmware	172
▼	To Update the B10n Application Software	172
▼	To Update the Software Noninteractively	174
▼	To Set the New Image to be the Default Image	175
▼	To Update the BSC Firmware	175
	Choosing the Boot Image	176
▼	To Specify and Make the Boot Image Permanent	176
▼	To Specify the Boot Image at Boot Time	177
	Updating Your Server	177
▼	To Update the SPARC Solaris Server Module	177
▼	To Update the x86 Solaris Server Module	178
▼	To Update the Linux Server Module	178
▼	To Upgrade the Linux Server Module From an Existing Installation	179
8.	Configuring a Sun Fire B100x Linux Server Blade	181

Sun Fire B100x Linux Server Blades	181
▼ To Configure a B100x Server Blade With Linux	181
▼ To Set Up a Sun Fire B100x/B200x Linux Server Blade	182
Online Documentation	186
Differences Between Linux Distributions.	186
A. Diagnostics and Troubleshooting	187
Diagnostic Tools	187
NPU POST and SDRAM Diagnostic Tests	188
Image Management and Troubleshooting	194
The Boot Process	194
Setting the Diagnostic Level	197
▼ To Set the Diagnostic Level	197
B. Tutorial and Examples	199
Exporting and Importing a Configuration	199
Exporting a Configuration	200
▼ To Export the Entire Configuration	200
▼ To Import the Entire Configuration	201
Configuring Layer 4 and Layer 7 Load Balancing	202
Setting the Networking Configurations	202
▼ To Configure the IP Addresses	202
▼ To Configure the Default Gateway	202
▼ To Configure the DNS Server	202
▼ To Configure DNS Suffix	203
Configuring a Basic Layer 4 Service Without Rules	203
▼ To Create a Layer 4 Service	203
▼ To Add Two blade servers to the Default Load Balancing Group for the Service	203
▼ To Add Another Server to the Default Load Balancing Group	204

▼ To Remove a Server From the Default Load Balancing Group	204
Configuring a Basic Layer 4 Service With Rules	205
▼ To Configure a Layer 4 Service with Layer 4 (IP) Rule	205
Configuring a Basic Layer 7 Service	207
▼ To Configure a Basic Layer 7 Service Without Rules	207
▼ To Configure a Layer 7 Service with Layer 7 Rules	207
Configuring a Layer 7 Service with SSL	210
▼ To Create SSL Devices	211
▼ To Create a Load Balancing Service with SSL	212
▼ To Add SSL Device to Service	212
▼ To Configure a Service for IP Persistence	214
▼ To Remove IP Persistence From a Service	214
▼ To Configure a Service for Tracking	215
▼ To Remove Port Tracking	215
▼ To Add an End Point Tracking to a Service	215
▼ To Remove End Point Tracking	216
▼ To Configure a Service for Cookie-Based Persistence	216
▼ To Remove Cookie Persistence from a Service	217
▼ To Configure a UDP Service	217
▼ To Configure an FTP Service	218
▼ To Add an End Point to a Service to Make it Multi-homed	220
Setting Up VLAN	221
Set Up on the Switch	221
▼ To Set Up VLAN on the Server	221
▼ To Set Up VLAN on a Load Balancing Blade	223
Configuring Failover	224
Preparation of Load Balancing Blades	224
Configuring Basic Path Failover	225

- ▼ To Add a Path Failover Target Address to Interface 0 225
- ▼ To Add a Path Failover Target Address to Interface 1 225
- ▼ To Enable Path Failover Monitoring 225
- ▼ To Configure Path Failover Monitoring Parameters 225
- ▼ To Show the Path Failover Status 226
- ▼ To Disable Path Failover Monitoring 226
- ▼ To Remove a Path Failover Target Address on Interface 0 227
- ▼ To Remove a Path Failover Target Address on Interface 1 227

Configuring Basic Blade Failover 227

Minimum Required Commands 227

- ▼ To Set Up the Peer IP Addresses on Both Blades 227
- ▼ To Enable Failover Monitoring on Both Blades 228
- ▼ To Start Failover Synchronization on Both Blades 228
- ▼ To Show the Configured Failover Information on Both Sides 229
- ▼ To Disable Failover Monitoring on Either Blade 230
- ▼ To Stop Failover Synchronization on Both Blades 231
- ▼ To Set Up the Failover Monitoring Parameters on the Active Blade 231
- ▼ To Force the Standby to Active on the Active Blade 231
- ▼ To Sync Up the Configurations on the Active Blade 231
- ▼ To Remove the Failover State File on Either Blade 231
- ▼ To Remove the Running Load Balancing Configuration on Either Blade 232
- ▼ To Remove the Failover Configuration on Either Blade 232

Failover Synchronization and the `commit` Command 232

Displaying Failover Module Information 233

- ▼ To Dump Monitoring Information 233
- ▼ To Dump the Ramdisk Directory 233
- ▼ To Dump Failover Information 233

- ▼ To Dump the Failover Synchronization Task 233
- ▼ To Dump the Failover Monitoring Task 233

C. Enterprise VLANs 235

- Enterprise Example Deploying VLANs 235
 - User Access Control 238
 - Definition of VLAN Terms 238
- System Components and VLAN Attributes 239
 - B1600 Switch 239
 - B10n Content Load Balancer 239
 - B10p SSL Proxy 239
 - Server Blade 240
- Enterprise Example 240
 - SSC VLAN Configuration 242
 - The Loadbalancer VLAN Configuration 245
 - The SSL Proxy Blade Configuration 246
 - The Solaris Server VLAN configuration 248

D. Alphabetical Command Reference 251

- A 251
- B 251
- C 251
- D 252
- E 253
- F 253
- H 254
- I 254
- L 254
- M 255

N	255
P	256
R	256
S	257
T	259
U	259
V	259
W	260

E. Scripting 261

F. Load Balancing Terms 265

Sun Fire B10n Load Balancing Terms 265

Glossary 267

Index 271

Figures

FIGURE 1-1	Ethernet Ports and Interfaces on the B1600 System Chassis and their Default VLAN Numbers	11
FIGURE 1-2	Dedicated Management Network and Web Server Network Isolated from the Backend Network	12
FIGURE 1-3	Load Balancing Group Configuration	14
FIGURE 1-4	Sample Topology: Dual Tree Using External and Internal Switches	15
FIGURE 2-1	The Filler Panel Locking Mechanism	22
FIGURE 2-2	Disengaging the Blade-Locking Mechanism	22
FIGURE 2-3	Removing the Filler Panel	23
FIGURE 2-4	Blade Locking Mechanism	23
FIGURE 2-5	Aligning and Inserting the Blade	24
FIGURE 2-6	Closing the Blade Lever Mechanism	25
FIGURE 2-7	External Cable Connections	27
FIGURE 2-8	The 10/100/1000BASE-T Data Network Ports	28
FIGURE 2-9	Serial Port Pin Numbers	29
FIGURE C-1	Enterprise Security Model	237
FIGURE C-2	Enterprise Example	241

Tables

TABLE 1-1	Hardware and Software Requirements	3
TABLE 1-2	Supported Servers and Operating Systems	5
TABLE 1-3	Zip Files Available at the Sun Download Center	6
TABLE 1-4	VLAN Based Security	18
TABLE 2-1	Blade and Power Supply Status Codes	26
TABLE 2-2	10/100/1000BASE-T Data Network Port Pinouts	28
TABLE 2-3	Serial Port Pinouts	29
TABLE 4-1	<code>user</code> Commands	48
TABLE 4-2	Parameter Description for User Access	50
TABLE 4-3	Parameters for Setting the IP Address	52
TABLE 4-4	Parameters for Sending a <code>ping</code> Request	54
TABLE 4-5	Parameters and Variables for Setting Up a Server for Monitoring	59
TABLE 4-6	Parameters for Setting Up Application Monitoring	59
TABLE 4-7	Parameters for Configuring Application Monitoring	60
TABLE 4-8	Parameters for Adding SSL Device Configurations	62
TABLE 4-9	Parameters for Adding and Removing Port Pairs	64
TABLE 4-10	Parameters for Adding or Removing an SSL Device Interface	65
TABLE 4-11	Options for Setting the TCP Parameters	68
TABLE 4-12	Parameters for Creating a Load Balancing Service	71
TABLE 4-13	Parameters for Configuring the Subnet Mask for a VIP	73

TABLE 4-14	Parameters for Adding SSL Devices to a Service	74
TABLE 4-15	Parameters for Removing SSL Devices from a Service	75
TABLE 4-16	Parameters for Modifying SSL Devices in a Service	76
TABLE 4-17	Parameters for Adding a Default Load Balancing Group to a Service	77
TABLE 4-18	TCP, Parameters for a Service	79
TABLE 4-19	Service Point Parameters	81
TABLE 4-20	Service Point Parameters	82
TABLE 4-21	Parameters for Configuring a Service for Persistence	84
TABLE 4-22	Parameters for Configuring a Service for Tracking	86
TABLE 4-23	Parameters for Configuring a Service for Cookie-Based Persistence	88
TABLE 4-24	Parameters for Enabling or Disabling a Server	92
TABLE 4-25	Parameters for Creating an IP Load Balancing Rule	94
TABLE 4-26	Parameters for Creating an HTTP Load Balancing Rule	95
TABLE 4-27	Parameters for Creating Load Balancing Groups	100
TABLE 4-28	Parameters for Adding Rules to a Load Balancing Group	102
TABLE 4-29	Parameters for Adding Servers to a LB Group	104
TABLE 4-30	Parameters for Removing Servers to a LB Group	105
TABLE 4-31	Parameters for Setting Servers to Active or Standby	106
TABLE 4-32	Parameters for Modifying a Load Balancing Group Scheme	107
TABLE 4-33	Parameters for Modifying a Load Balancing Group Scheme	108
TABLE 4-34	Parameters for Configuring the Diagnostic and Verbosity Level	114
TABLE 4-35	Parameters for Configuring the Debug Level for Specific Modules	115
TABLE 4-36	Module Names to Use for Configuring the Debug Level	115
TABLE 4-37	Parameters for the <code>dump memory</code> Command	119
TABLE 4-38	Parameters for the <code>dump module</code> Command	120
TABLE 4-39	Module Names to Use with the <code>dump module</code> Command	120
TABLE 4-40	Modules in the NPU Listed by Index	121
TABLE 4-41	Tasks Listed by Index for the <code>dump module task</code> Command	122
TABLE 4-42	VxWorks Network Statistics/Tables by Index	123
TABLE 4-43	Failover Sub-modules by Index	124

TABLE 4-44	Statistics Listed by Index for the <code>dump module stats</code> Command	124
TABLE 5-1	List of Configuration Commands	156
TABLE 6-1	Parameters for the <code>no enable vlan</code> Command	162
TABLE B-1	Files and Directories to be Compressed and Exported	200

Regulatory Compliance Statements

Your Sun product is marked to indicate its compliance class:

- Federal Communications Commission (FCC) — USA
- Industry Canada Equipment Standard for Digital Equipment (ICES-003) — Canada
- Voluntary Control Council for Interference (VCCI) — Japan
- Bureau of Standards Metrology and Inspection (BSMI) — Taiwan

Please read the appropriate section that corresponds to the marking on your Sun product before attempting to install the product.

FCC Class A Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, 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 it is not installed and used in accordance with the instruction manual, it 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 will be required to correct the interference at his own expense.

Shielded Cables: Connections between the workstation and peripherals must be made using shielded cables to comply with FCC radio frequency emission limits. Networking connections can be made using unshielded twisted-pair (UTP) cables.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

FCC Class B Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/television technician for help.

Shielded Cables: Connections between the workstation and peripherals must be made using shielded cables in order to maintain compliance with FCC radio frequency emission limits. Networking connections can be made using unshielded twisted pair (UTP) cables.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

ICES-003 Class A Notice - Avis NMB-003, Classe A

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

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

ICES-003 Class B Notice - Avis NMB-003, Classe B

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

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


VCCI 基準について

クラス A VCCI 基準について

クラス A VCCI の表示があるワークステーションおよびオプション製品は、クラス A 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

クラス B VCCI 基準について

クラス B VCCI の表示  があるワークステーションおよびオプション製品は、クラス B 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス B 情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。取扱説明書に従って正しい取り扱いをしてください。

BSMI Class A Notice

The following statement is applicable to products shipped to Taiwan and marked as Class A on the product compliance label.

警告使用者：

這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。

Safety Agency Compliance Statements

Read this section before beginning any procedure. The following text provides safety precautions to follow when installing a Sun Microsystems product.

Safety Precautions

For your protection, observe the following safety precautions when setting up your equipment:

- Follow all cautions and instructions marked on the equipment.
- Ensure that the voltage and frequency of your power source match the voltage and frequency inscribed on the equipment's electrical rating label.
- Never push objects of any kind through openings in the equipment. Dangerous voltages may be present. Conductive foreign objects could produce a short circuit that could cause fire, electric shock, or damage to your equipment.

Symbols

The following symbols may appear in this book:



Caution – There is a risk of personal injury and equipment damage. Follow the instructions.



Caution – Hot surface. Avoid contact. Surfaces are hot and may cause personal injury if touched.



Caution – Hazardous voltages are present. To reduce the risk of electric shock and danger to personal health, follow the instructions.



On – Applies AC power to the system.

Depending on the type of power switch your device has, one of the following symbols may be used:



Off – Removes AC power from the system.



Standby – The On/Standby switch is in the standby position.

Modifications to Equipment

Do not make mechanical or electrical modifications to the equipment. Sun Microsystems is not responsible for regulatory compliance of a modified Sun product.

Placement of a Sun Product



Caution – Do not block or cover the openings of your Sun product. Never place a Sun product near a radiator or heat register. Failure to follow these guidelines can cause overheating and affect the reliability of your Sun product.



Caution – The workplace-dependent noise level defined in DIN 45 635 Part 1000 must be 70Db(A) or less.

SELV Compliance

Safety status of I/O connections comply to SELV requirements.

Power Cord Connection



Caution – Sun products are designed to work with single-phase power systems having a grounded neutral conductor. To reduce the risk of electric shock, do not plug Sun products into any other type of power system. Contact your facilities manager or a qualified electrician if you are not sure what type of power is supplied to your building.



Caution – Not all power cords have the same current ratings. Household extension cords do not have overload protection and are not meant for use with computer systems. Do not use household extension cords with your Sun product.



Caution – Your Sun product is shipped with a grounding type (three-wire) power cord. To reduce the risk of electric shock, always plug the cord into a grounded power outlet.

The following caution applies only to devices with a Standby power switch:



Caution – The power switch of this product functions as a standby type device only. The power cord serves as the primary disconnect device for the system. Be sure to plug the power cord into a grounded power outlet that is nearby the system and is readily accessible. Do not connect the power cord when the power supply has been removed from the system chassis.

Lithium Battery



Caution – On Sun CPU boards, there is a lithium battery molded into the real-time clock, SGS No. MK48T59Y, MK48TXXB-XX, MK48T18-XXXPCZ, M48T59W-XXXPCZ, or MK48T08. Batteries are not customer replaceable parts. They may explode if mishandled. Do not dispose of the battery in fire. Do not disassemble it or attempt to recharge it.

Battery Pack



Caution – There is a sealed lead acid battery in Sun Fire B10n blade units. Portable Energy Products No. TLC02V50. There is danger of explosion if the battery pack is mishandled or incorrectly replaced. Replace only with the same type of Sun Microsystems battery pack. Do not disassemble it or attempt to recharge it outside the system. Do not dispose of the battery in fire. Dispose of the battery properly in accordance with local regulations.

System Unit Cover

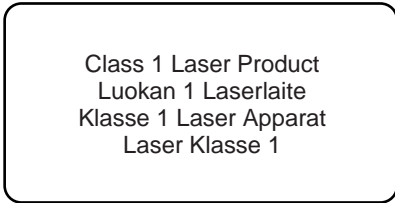
You must remove the cover of your Sun computer system unit to add cards, memory, or internal storage devices. Be sure to replace the top cover before powering on your computer system.



Caution – Do not operate Sun products without the top cover in place. Failure to take this precaution may result in personal injury and system damage.

Laser Compliance Notice

Sun products that use laser technology comply with Class 1 laser requirements.

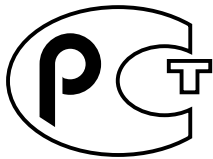


CD-ROM



Caution – Use of controls, adjustments, or the performance of procedures other than those specified herein may result in hazardous radiation exposure.

GOST-R Certification Mark



Conformité aux normes de sécurité

Lisez attentivement la section suivante avant de commencer la procédure. Le document ci-dessous présente les consignes de sécurité à respecter au cours de l'installation d'un produit Sun Microsystems.

Mesures de sécurité

Pour votre protection, observez les mesures de sécurité suivantes lors de l'installation de l'équipement:

- Observez tous les avertissements et consignes indiqués sur l'équipement.
- Assurez-vous que la tension et la fréquence de votre source d'alimentation électrique correspondent à la tension et à la fréquence indiquées sur l'étiquette de la tension électrique nominale du matériel.

- N'insérez en aucun cas un objet quelconque dans les orifices de l'équipement. Des tensions potentiellement dangereuses risquent d'être présentes dans l'équipement. Tout objet étranger conducteur risque de produire un court-circuit pouvant présenter un risque d'incendie ou de décharge électrique, ou susceptible d'endommager le matériel.

Symboles

Les symboles suivants peuvent figurer dans cet ouvrage :



Attention – Vous risquez d'endommager le matériel ou de vous blesser. Observez les consignes indiquées.



Attention – Surface brûlante. Evitez tout contact. Ces surfaces sont brûlantes. Vous risquez de vous blesser si vous les touchez.



Attention – Tensions dangereuses. Pour réduire les risques de décharge électrique et de danger physique, observez les consignes indiquées.



Marche – Met le système sous tension alternative.

Selon le type d'interrupteur marche/arrêt dont votre appareil est équipé, l'un des symboles suivants sera utilisé :



Arrêt – Met le système hors tension alternative.



Veilleuse – L'interrupteur Marche/Veille est sur la position de veille.

Modifications de l'équipement

N'apportez aucune modification mécanique ou électrique à l'équipement. Sun Microsystems décline toute responsabilité quant à la non-conformité éventuelle d'un produit Sun modifié.

Positionnement d'un produit Sun



Attention – N'obstruez ni ne recouvrez les orifices de votre produit Sun. N'installez jamais un produit Sun près d'un radiateur ou d'une source de chaleur. Si vous ne respectez pas ces consignes, votre produit Sun risque de surchauffer et son fonctionnement en sera altéré.



Attention – Le niveau de bruit inhérent à l'environnement de travail, tel qu'il est défini par la norme DIN 45 635 - section 1000, doit être inférieur ou égal à 70Db(A).

Conformité aux normes SELV

Le niveau de sécurité des connexions E/S est conforme aux normes SELV.

Raccordement à la source d'alimentation électrique



Attention – Les produits Sun sont conçus pour fonctionner avec des systèmes d'alimentation électrique monophasés avec prise de terre. Pour réduire les risques de décharge électrique, ne branchez jamais les produits Sun sur une source d'alimentation d'un autre type. Contactez le gérant de votre bâtiment ou un électricien agréé si vous avez le moindre doute quant au type d'alimentation fourni dans votre bâtiment.



Attention – Tous les cordons d'alimentation n'ont pas la même intensité nominale. Les cordons d'alimentation à usage domestique ne sont pas protégés contre les surtensions et ne sont pas conçus pour être utilisés avec des ordinateurs. N'utilisez jamais de cordon d'alimentation à usage domestique avec les produits Sun.



Attention – Votre produit Sun est livré avec un cordon d'alimentation avec raccord à la terre (triphase). Pour réduire les risques de décharge électrique, branchez toujours ce cordon sur une source d'alimentation mise à la terre.

L'avertissement suivant s'applique uniquement aux systèmes équipés d'un interrupteur Veille :



Attention – L'interrupteur d'alimentation de ce produit fonctionne uniquement comme un dispositif de mise en veille. Le cordon d'alimentation constitue le moyen principal de déconnexion de l'alimentation pour le système. Assurez-vous de le brancher dans une prise d'alimentation mise à la terre près du système et facile d'accès. Ne le branchez pas lorsque l'alimentation électrique ne se trouve pas dans le châssis du système.

Pile au lithium



Attention – Sur les cartes UC Sun, une batterie au lithium a été moulée dans l'horloge temps réel, de type SGS n° MK48T59Y, MK48TXXB-XX, MK48T18-XXXPCZ, M48T59W-XXXPCZ ou MK48T08. Cette batterie ne doit pas être remplacée par le client. Elle risque d'exploser en cas de mauvaise manipulation. Ne la jetez pas au feu. Ne la démontez pas et ne tentez pas de la recharger.

Bloc-batterie



Attention – Les unités Sun Fire B10n blade contiennent une batterie étanche au plomb. Produits énergétiques portatifs n° TLC02V50. Il existe un risque d'explosion si ce bloc batterie est manipulé ou installé de façon incorrecte. Ne le remplacez que par un bloc batterie Sun Microsystems du même type. Ne le démontez pas et n'essayez pas de le recharger hors du système. Ne le jetez pas au feu. Mettez-le au rebut conformément aux réglementations locales en vigueur.

Couvercle du système

Pour ajouter des cartes, de la mémoire ou des unités de stockage internes, vous devez démonter le couvercle de votre système Sun. N'oubliez pas de le remettre en place avant de mettre le système sous tension.



Attention – Ne travaillez jamais avec un produit Sun dont le couvercle n'est pas installé. Si vous ne respectez pas cette consigne, vous risquez de vous blesser ou d'endommager le système.

Avis de conformité des appareils laser

Les produits Sun faisant appel à la technologie laser sont conformes aux normes de sécurité des appareils laser de classe 1.

Class 1 Laser Product
Luokan 1 Laserlaite
Klasse 1 Laser Apparat
Laser Klasse 1

CD-ROM



Attention – L'utilisation de contrôles et de réglages ou l'application de procédures autres que ceux spécifiés dans le présent document peuvent entraîner une exposition à des radiations dangereuses.

Notice de qualité GOST-R



Einhaltung sicherheitsbehördlicher Vorschriften

Lesen Sie diesen Abschnitt sorgfältig durch, bevor Sie mit dem Arbeitsablauf beginnen. Der folgende Text beschreibt Sicherheitsmaßnahmen, die bei der Installation von Sun-Produkten zu beachten sind.

Sicherheitsmaßnahmen

Zu Ihrem eigenen Schutz sollten Sie die folgenden Sicherheitsmaßnahmen bei der Installation befolgen:

- Befolgen Sie alle auf die Geräte aufgedruckten Anweisungen und Warnhinweise.
- Beachten Sie die Geräteaufschrift, um sicherzustellen, daß Netzspannung und -frequenz mit der Gerätespannung und -frequenz übereinstimmen.
- Führen Sie niemals Gegenstände in die Geräteöffnungen ein. Es könnten elektrische Spannungsfelder vorhanden sein. Leitende Fremdkörper können Kurzschlüsse, Feuer und elektrische Schläge verursachen oder Ihr Gerät beschädigen.

Symbole

Die folgenden Symbole werden in diesem Handbuch verwendet:



Achtung – Es besteht die Gefahr der Verletzung und der Beschädigung des Geräts. Befolgen Sie die Anweisungen.



Achtung – Heiße Oberfläche. Vermeiden Sie jede Berührung. Diese Oberflächen sind sehr heiß und können Verbrennungen verursachen.



Achtung – Elektrisches Spannungsfeld vorhanden. Befolgen Sie die Anweisungen, um elektrische Schläge und Verletzungen zu vermeiden.



Ein – Das System wird mit Wechselstrom versorgt.

Abhängig von der Art des Stromschalters Ihres Gerätes wird eventuell eines der folgenden Symbole verwendet:



Aus – Das System wird nicht mehr mit Wechselstrom versorgt.



Wartezustand – (Der Ein-/Standby-Schalter befindet sich in der Standby-Position).

Modifikationen des Geräts

Nehmen Sie keine elektrischen oder mechanischen Gerätemodifikationen vor. Sun Microsystems ist für die Einhaltung der Sicherheitsvorschriften von modifizierten Sun-Produkten nicht haftbar.

Aufstellung von Sun-Geräten



Achtung – Geräteöffnungen Ihres Sun-Produkts dürfen nicht blockiert oder abgedeckt werden. Sun-Geräte sollten niemals in der Nähe von Heizkörpern oder Heißluftklappen aufgestellt werden. Nichtbeachtung dieser Richtlinien können Überhitzung verursachen und die Zuverlässigkeit Ihres Sun-Geräts beeinträchtigen.



Achtung – Der Geräuschpegel, definiert nach DIN 45 635 Part 1000, darf am Arbeitsplatz 70dB(A) nicht überschreiten.

SELV-Richtlinien

Alle Ein-/Ausgänge erfüllen die SELV-Anforderungen.

Netzanschlußkabel



Achtung – Sun-Geräte benötigen ein einphasiges Stromversorgungssystem mit eingebautem Erdleiter. Schließen Sie Sun-Geräte nie an ein anderes Stromversorgungssystem an, um elektrische Schläge zu vermeiden. Falls Sie die Spezifikationen der Gebäudestromversorgung nicht kennen, sollten Sie den Gebäudeverwalter oder einen qualifizierten Elektriker konsultieren.



Achtung – Nicht alle Netzanschlußkabel besitzen die gleiche Stromleitung. Normale Verlängerungskabel besitzen keinen Überspannungsschutz und sind nicht für den Gebrauch mit Computersystemen geeignet. Benutzen Sie keine Haushaltverlängerungskabel für Sun-Geräte.



Achtung – Ihr Sun-Gerät wurde mit einem geerdeten (dreiadrigen) Netzanschlußkabel geliefert. Stecken Sie dieses Kabel immer nur in eine geerdete Netzsteckdose, um Kurzschlüsse zu vermeiden.

Der folgende Hinweis bezieht sich nur auf Geräte mit Standby-Stromschalter:



Achtung – Der Stromschalter dieses Produkts funktioniert nur als Standby-Gerät. Das Netzanschlußkabel dient als Hauptabschaltgerät für das System. Stellen Sie sicher, daß Sie das Netzanschlußkabel in den geerdeten Stromausgang in der Nähe des Systems einstecken. Schließen Sie das Netzanschlußkabel nicht an, wenn die Stromzufuhr vom Systemgehäuse entfernt wurde.

Lithium-Batterie



Achtung – CPU-Karten von Sun verfügen über eine Echtzeituhr mit integrierter Lithiumbatterie, Teile-Nr. MK48T59Y, MK48TXXB-XX, MK48T18-XXXPCZ, M48T59W-XXXPCZ oder MK48T08. Batterien sollten nicht vom Kunden ausgetauscht werden. Sie können bei falscher Handhabung explodieren. Entsorgen Sie die Batterien nicht im Feuer. Entfernen Sie sie nicht und versuchen Sie auch nicht, sie wiederaufzuladen.

Batterien



Achtung – Die Geräte Sun Fire B10n blade enthalten auslaufsichere Bleiakumulatoren, Produkt-Nr. TLC02V50 für portable Stromversorgung. Wenn die Batterien nicht richtig gehandhabt oder ausgetauscht werden, besteht Explosionsgefahr. Tauschen Sie Batterien nur gegen Batterien gleichen Typs von Sun Microsystems aus. Versuchen Sie nicht, die Batterien zu entfernen oder außerhalb des Geräts wiederaufzuladen. Entsorgen Sie die Batterien nicht im Feuer. Entsorgen Sie die Batterien ordnungsgemäß entsprechend den vor Ort geltenden Vorschriften.

Abdeckung des Systems

Sie müssen die Abdeckung des Sun-Computersystems entfernen, um Karten, Speicher oder interne Speichergeräte hinzuzufügen. Stellen Sie sicher, daß Sie die Abdeckung wieder einsetzen, bevor Sie den Computer einschalten.



Achtung – Sun-Geräte dürfen nicht ohne Abdeckung in Gebrauch genommen werden. Nichtbeachtung dieses Warnhinweises kann Verletzungen oder Systembeschädigungen zur Folge haben.

Laserrichtlinien

Alle Sun-Produkte, die Lasertechnologie nutzen, erfüllen die Laserrichtlinien der Klasse 1.

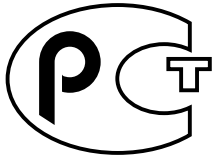
Class 1 Laser Product
Luokan 1 Laserlaite
Klasse 1 Laser Apparät
Laser Klasse 1

CD-ROM



Achtung – Die Verwendung von anderen Steuerungen und Einstellungen oder die Durchführung von Arbeitsabläufen, die von den hier beschriebenen abweichen, können gefährliche Strahlungen zur Folge haben.

Verbandsmarke GOST-R



Normativas de seguridad

Lea esta sección antes de llevar a cabo cualquier procedimiento. El texto que aparece a continuación explica las medidas de seguridad que deben tomarse durante la instalación de un producto Sun Microsystems.

Medidas de seguridad

Por su propia seguridad, tome las medidas de seguridad siguientes al instalar el equipo:

- Siga todas las avisos y las instrucciones que aparecen impresas en el equipo.
- Cerciórese de que el voltaje y la frecuencia de la fuente de alimentación coinciden con el voltaje y frecuencia indicados en la etiqueta de clasificación eléctrica del equipo.
- No introduzca objetos de ningún tipo a través de las aberturas del equipo. Dentro pueden darse voltajes peligrosos. Los objetos conductores extraños podrían producir un cortocircuito y, en consecuencia, fuego, descargas eléctricas o daños en el equipo.

Símbolos

Los símbolos siguientes pueden aparecer en este manual:



Precaución – Existe el riesgo de que se produzcan lesiones personales y daños en el equipo. Siga las instrucciones.



Precaución – Superficie caliente. Evite todo contacto. Las superficies están calientes y pueden causar lesiones personales si se tocan.



Precaución – Riesgo de voltajes peligrosos. Para reducir el riesgo de descargas eléctricas y de daños en la salud de las personas, siga las instrucciones.



Encendido – Proporciona alimentación de CA al sistema.

Según el tipo de interruptor de alimentación del que disponga el dispositivo, se utilizará uno de los símbolos siguientes:



Apagado – Corta la alimentación de CA del sistema.



Espera – El interruptor de encendido/espera está en la posición de espera.

Modificaciones en el equipo

No realice modificaciones mecánicas ni eléctricas en el equipo. Sun Microsystems no se hará responsable del cumplimiento de las normas en el caso de un producto Sun que ha sido modificado.

Lugar y colocación de un producto Sun



Precaución – No obstruya ni tape las rejillas del producto Sun. Nunca coloque un producto Sun cerca de radiadores o fuentes de calor. El incumplimiento de estas directrices puede causar un recalentamiento y repercutir en la fiabilidad del producto Sun.



Precaución – El nivel de ruido en el lugar de trabajo, definido en el apartado 1000 de DIN 45 635, debe ser 70 Db (A) o inferior.

Cumplimiento de las normas SELV

Las condiciones de seguridad de las conexiones de E/S cumplen las normas SELV.

Conexión del cable de alimentación



Precaución – Los productos Sun han sido diseñados para funcionar con sistemas de alimentación monofásicos que tengan un conductor neutral a tierra. Para reducir el riesgo de descargas eléctricas, no enchufe ningún producto Sun a otro tipo de sistema de alimentación. Si no está seguro del tipo de alimentación del que se dispone en el edificio, póngase en contacto con el encargado de las instalaciones o con un electricista calificado.



Precaución – No todos los cables de alimentación tienen la misma clasificación de corriente. Los cables de prolongación domésticos no ofrecen protección frente a sobrecargas y no están diseñados para ser utilizados con sistemas informáticos. No utilice cables de prolongación domésticos con el producto Sun.



Precaución – El producto Sun se suministra con un cable de alimentación (de tres hilos) con conexión a tierra. Para reducir el riesgo de descargas eléctricas, enchufe siempre el cable a una toma de corriente con conexión a tierra.

La precaución siguiente sólo se aplica a aquellos dispositivos que posean un interruptor de alimentación de espera:



Precaución – El interruptor de alimentación del producto funciona como dispositivo de espera solamente. El cable de alimentación actúa como el dispositivo de desconexión primario del sistema. Cerciérese de enchufar el cable de alimentación a una toma de corriente con conexión a tierra situada cerca del sistema y a la que se pueda acceder con facilidad. No conecte el cable de alimentación cuando se haya quitado la fuente de alimentación del bastidor del sistema.

Batería de litio



Precaución – En la placa CPU de los productos Sun, hay una batería de litio incorporada en el reloj en tiempo real, SGS núm. MK48T59Y, MK48TXXB-XX, MK48T18-XXXPCZ, M48T59W-XXXPCZ o MK48T08. Los usuarios no deben cambiar las baterías. Podrían estallar si no se utilizan adecuadamente. No arroje la batería al fuego. No la desmonte ni intente recargarla.

Paquete de baterías



Precaución – Las unidades Sun Fire B10n blade contienen una batería de plomo sellada, Productos eléctricos portátiles núm. TLC02V50. Existe el riesgo de explosión si el paquete de baterías no se utiliza correctamente o se sustituye de forma incorrecta. Sustitúyalo sólo por el mismo tipo de paquete de baterías de Sun Microsystems. No lo desmonte o intente recargarlo fuera del sistema. No arroje la batería al fuego. Deshágase de las baterías correctamente siguiendo las normas locales vigentes.

Cubierta de la unidad del sistema

Debe retirar la cubierta de la unidad del sistema informático Sun para añadir tarjetas, memoria o dispositivos de almacenamiento internos. Asegúrese de volver a colocar la cubierta superior antes de encender el equipo.



Precaución – No ponga en funcionamiento los productos Sun sin que la cubierta superior se encuentre instalada. De lo contrario, podrían producirse lesiones personales o daños en el sistema.

Aviso de cumplimiento de las normas para láser

Los productos Sun que utilizan tecnología láser cumplen los requisitos para láser de Clase 1.

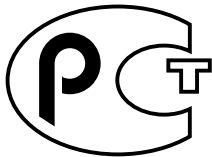
Class 1 Laser Product
Luokan 1 Laserlaite
Klasse 1 Laser Apparat
Laser Klasse 1

CD-ROM



Precaución – La utilización de controles, ajustes o la realización de los procedimientos distintos a los especificados en el presente documento podrían provocar la exposición a radiaciones peligrosas.

Certificación GOST-R



Nordic Lithium Battery Cautions

Norge



Advarsel – Litiumbatteri — Eksplosjonsfare. Ved utskifting benyttes kun batteri som anbefalt av apparatfabrikanten. Brukt batteri returneres apparatleverandøren.

Sverige



Varning – Explosionsfara vid felaktigt batteribyte. Använd samma batterityp eller en ekvivalent typ som rekommenderas av apparattillverkaren. Kassera använt batteri enligt fabrikantens instruktion.

Danmark



Advarsel! – Litiumbatteri — Eksplosionsfare ved fejlagtig håndtering. Udskiftning må kun ske med batteri af samme fabrikat og type. Levér det brugte batteri tilbage til leverandøren.

Suomi



Varoitus – Paristo voi räjähtää, jos se on virheellisesti asennettu. Vaihda paristo ainoastaan laitevalmistajan suosittelemaan tyyppiin. Hävitä käytetty paristo valmistajan ohjeiden mukaisesti.

Declaration of Conformity

Compliance Model Number: BP-4
Product Family Name: Sun Fire B10n blade

EMC

USA—FCC Class A

This equipment complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This equipment may not cause harmful interference.
2. This equipment must accept any interference that may cause undesired operation.

European Union

This equipment complies with the following requirements of the EMC Directive 89/336/EEC:

As Telecommunication Network Equipment (TNE) in Both Telecom Centers and Other Than Telecom Centers per (as applicable):

EN300-386 V.1.3.1 (09-2001) Required Limits:

EN55022/CISPR22	Class A
EN61000-3-2	Pass
EN61000-3-3	Pass
EN61000-4-2	6 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m 80-1000MHz, 10 V/m 800-960 MHz, and 1400-2000 MHz
EN61000-4-4	1 kV AC and DC Power Lines, 0.5 kV Signal Lines
EN61000-4-5	2 kV AC Line-Gnd, 1 kV AC Line-Line and Outdoor Signal Lines, 0.5 kV Indoor signal Lines > 10m
EN61000-4-6	3 V
EN61000-4-11	Pass

As Information Technology Equipment (ITE) Class A per (as applicable):

EN55022:1998/CISPR22:1997 Class A

EN55024:1998 Required Limits:

EN61000-4-2	4 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m
EN61000-4-4	1 kV AC Power Lines, 0.5 kV Signal and DC Power Lines
EN61000-4-5	1 kV AC Line-Line and Outdoor Signal Lines, 2 kV AC Line-Gnd, 0.5 kV DC Power Lines
EN61000-4-6	3 V
EN61000-4-8	1 A/m
EN61000-4-11	Pass

EN61000-3-2:1995 + A1, A2, A14 Pass

EN61000-3-3:1995 Pass

Safety: This equipment complies with the following requirements of the Low Voltage Directive 73/23/EEC:

EC Type Examination Certificates:

EN60950:2000, 3rd Edition TÜV Rheinland Certificate No. xxxxxxxxxxxxxx

IEC 60950:2000, 3rd Edition CB Scheme Certificate No. xxxxxxxxxxxxxx

Evaluated to all CB Countries

UL 60950, 3rd Edition, CSA C22.2 No. 60950-00 File: Vol. Sec.

UL 60950, 3rd Edition, CSA C22.2 No. 950-00 File: Vol. Sec.

FDA DHHS Accession Number (Monitors Only)

Supplementary Information: This product was tested and complies with all the requirements for the CE Mark.

/S/

Dennis P. Symanski
Manager, Compliance Engineering
Sun Microsystems, Inc.
4150 Network Circle, MPK15-102
Santa Clara, CA 95054 U.S.A.
Tel: 650-786-3255
Fax: 650-786-3723

DATE

/S/

Pamela J. Dullaghan
Quality Program Manager
Sun Microsystems Scotland, Limited
Springfield, Linlithgow
West Lothian, EH49 7LR
Scotland, United Kingdom
Tel: +44 1 506 672 395 Fax: +44 1 506 670 011

DATE

Preface

The Sun Fire B10n Content Load Balancing Blade Version 1.3 Administration Guide provides installation and configuration instructions for the Sun Fire™ content load balancing blade. These instructions are designed for an experienced system administrator with networking knowledge.

How This Book Is Organized

Chapter 1 describes the product hardware and software and lists hardware and software requirements and features.

Chapter 2 describes how to install the hardware.

Chapter 3 describes the procedures for preparing the blade for load balancing.

Chapter 4 describes the management and control interfaces available through the Sun Fire B10n blade command line interface (CLI). The chapter lists the CLI commands under the various management categories with the appropriate parameters.

Chapter 5 describes how to prepare the system for failover.

Chapter 6 describes how to configure the VLAN parameters.

Chapter 7 describes how to download and install firmware upgrades. The chapter includes the URL for upgrades.

Chapter 8 describes how to configure the Sun Fire B10n blade with the Linux operating system.

Appendix A contains an overview of the diagnostic tools and instructions for using them. There is also a section outlining some common troubleshooting issues.

Appendix B provides a tutorial for configuring the product, including examples.

Appendix D provides an alphabetical listing of all the commands for the Sun Fire B10n blade.

Using UNIX Commands

This document might not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See the following for this information:

- Software documentation that you received with your system
- Solaris™ operating environment documentation, which is at <http://docs.sun.com>

Typographic Conventions

Typeface*	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with on-screen computer output	% su Password:
<i>AaBbCc123</i>	Book titles, new words or terms, words to be emphasized. Replace command-line variables with real names or values.	Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. To delete a file, type rm <i>filename</i> .

* The settings on your browser might differ from these settings.

Shell Prompts

Shell	Prompt
C shell	<i>machine-name%</i>
C shell superuser	<i>machine-name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

Related Documentation

Application	Title	Part Number
Installation	<i>Sun Fire™ B1600 Blade System Chassis Hardware Installation Guide</i>	816-7614-10
Software setup	<i>Sun Fire B1600 Blade System Chassis Software Setup Guide</i>	816-3361-10
Safety and compliance	<i>Sun Fire B1600 Blade System Chassis Safety and Compliance Manual</i>	816-3364-10

Accessing Sun Documentation

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

<http://www.sun.com/documentation>

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in this document, go to:

<http://www.sun.com/service/contacting>

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

<http://www.sun.com/hwdocs/feedback>

Please include the title and part number of your document with your feedback:

Sun Fire B10n Content Load Balancing Blade Version 1.3 Administration Guide, part number 817-6210-10

Product Overview

This chapter describes the Sun Fire B10n blade hardware and software and lists both its features and the requirements for using it.

This chapter contains the following sections:

- “Hardware and Software Overview” on page 1
- “Software Architecture” on page 2
- “Supported Hardware and Software” on page 3
- “Product Features” on page 6
- “The Role of the Content Load Balancing Blade” on page 9
- “The Role of VLANs” on page 17
- “System Integration” on page 19
- “Usage Overview” on page 19

Refer to the latest issue of the Sun Fire™ Blade Application Journal for further configuration and architectural overview information. The Application Journal is available at: <http://processors.sfbay/ns/spblades/>

Hardware and Software Overview

The Sun Fire B10n blade is a networking product that provides content load balancing for Sun’s blade-based servers and other horizontally scaled Sun platforms. It is designed to work in the management framework of the Sun Fire™ B1600 blade system chassis. As part of the Sun Fire B1600 blade system chassis, the Sun Fire B10n blade connects to the Sun Fire B1600 blade system chassis midplane through two Gigabit Ethernet interfaces. The blade offers Layer 4 through Layer 7 load balancing. The server decision can be based on IP protocol, IP persistence, and TCP/UDP ports (Layer 4) or URLs, cookies, cookie persistence, and CGI scripts (Layer 7). Both these functions can operate up to the two Gigabit full-duplex line rate.

The Sun Fire B10n blade implements hardware assisted direct server to client response load balancing, which enables the switch capacity to be used for aggregate responses rather than individual link capacity to the blade. To enable this server to client direct data flow, you must install a software module (Server Module) on each server.

From a logical perspective, the following components make up a Sun Fire B10n blade:

- A Layer 4 parsing, classification, and forwarding engine.
- A Layer 7 content classifier that parses the request packet and matches the URL, CGI script, cookie persistence, or cookies. This unit also performs the TCP 3-way handshake, makes the load balancing decision, and hands the connection off to the server.
- Management software to set up and maintain service entry points, load balancing groups, and Layer 7 rule tables.

Software Architecture

The Sun Fire B10n blade provides optimized server to client response. To support this response and provide tight communications between the content load balancing blade and the B1600 blade servers a software module must be installed on each of these servers. This software module is referred to as the Server Module and is loaded using the Unix package add (`pkgadd`) process.

The content load balancing blade is based on specialized hardware including a general purpose microprocessor that runs a real time operating system. The code that runs on this processor is called the Application Software and can be updated using a TFTP process.

In addition to the general purpose processor there is a micro controller called the Blade Support Controller (BSC). The BSC is the primary interface to the Sun Fire B1600 Service Controllers (SC) and performs the Advanced Lights-out Management (ALOM) function for a given blade. These functions include powering on and off of the blades as well as monitoring functions. This is referred to as the BSC Firmware and can be updated using the “flashupdate” command which involves using TFTP.

The Sun Fire B10n software components:

- Server Module (`clbmod`)
- B10n Application Software
- BSC Firmware

Check the following web site to ensure you have the latest Sun Fire B10n software:

<http://www.sun.com/software/download/network.html>

The Sun Fire B10n blade has the capability to hold two versions of the Application Software and a diagnostic image. This allows a new image to be loaded without overwriting the active image. The blade must be rebooted to activate an image. See “Choosing the Boot Image” on page 176.

The B10n specialized hardware includes a rule based classification engine. The rules are entered through the command line interface and then compiled using a build process. See “Creating an HTTP Load Balancing Rule” on page 95.

Configuration of the load balancing blade can be achieved directly through the Command Line Interface (CLI) or by importing configuration files. The CLI interface is scriptable and access is through the B1600 chassis Service Controller (SC) interface. Connection to the SC can be direct RS232 or Telnet of the 10/100 ethernet port. (See Chapter 4.)

Supported Hardware and Software

Before using the Sun Fire B10n blade, ensure your system meets the hardware and software requirements listed in TABLE 1-1. If you need a server module for a later operating system release check the Sun Download Center at:

<http://www.sun.com/software/download/network.html>

TABLE 1-1 Hardware and Software Requirements

Hardware and Software	Requirements
Hardware	<ul style="list-style-type: none">• Sun Fire™ B10n content load balancing blade• Sun Fire B10p SSL proxy blade (optional)• Sun Fire B1600 blade system chassis• Sun Fire B100s blade server for SPARC or Sun Fire B100x/B200x blade servers for x86• Sun Fire V60/V65 and V210/V240 Servers (External to B1600 Chassis)
Software	<ul style="list-style-type: none">• Sun Fire B10n content load balancing blade application software 1.3.5 or subsequent compatible version• Sun Fire B10n content load balancing blade BSC (blade support control) firmware v5.1.4* or subsequent compatible version

TABLE 1-1 Hardware and Software Requirements (*Continued*)

Hardware and Software	Requirements
<i>Software (Continued)</i>	<ul style="list-style-type: none">• Sun Fire B100s Solaris Operating System versions:<ul style="list-style-type: none">Solaris 8 HW 12/02Solaris 8 HW 5/03Solaris 8 HW 7/03Solaris 8 HW 2/04Solaris 9 8/03Solaris 9 12/03Solaris 9 4/04• Sun Fire V210/V240 Operating System versions:<ul style="list-style-type: none">Solaris 8 HW 12/02Solaris 8 HW 5/03Solaris 8 HW 7/03Solaris 8 HW 2/04Solaris 9 8/03Solaris 9 12/03Solaris 9 4/04• Sun Fire B100x/B200x Operating System versions:<ul style="list-style-type: none">x86 Solaris 9 12/03x86 Solaris 9 8/03x86 Solaris 9 4/04Red Hat Enterprise Linux (RHEL) Advanced Server 2.1Red Hat Enterprise Linux (RHEL) Advanced Server 2.1 Update 2Red Hat Enterprise Linux (RHEL) Advanced Server 3.0• Sun Fire V60/V65 Operating System versions:<ul style="list-style-type: none">Solaris 9 8/03Solaris 9 12/03Solaris 9 4/04Red Hat Enterprise Linux (RHEL) Advanced Server 2.1 Update 2Red Hat Enterprise Linux (RHEL) Advanced Server 3.0• Sun Fire B1600 SC (system controller) 1.2 or subsequent compatible system controller firmware

TABLE 1-1 Hardware and Software Requirements (*Continued*)

Hardware and Software	Requirements
	<ul style="list-style-type: none"> • B10n Solaris server module version v1.65 for Solaris, or B10n Linux server module version xxx1.41-1 for Red Hat Enterprise Linux Advanced Server 2.1.** • Sun GigaSwift Ethernet Adapter Patch ID 111883-18 or subsequent compatible patch for supported versions of the Solaris 8 software. Sun GigaSwift Ethernet Adapter Patch ID 112817-10 or subsequent compatible patch for supported versions of the Solaris 9 software.** • Sun Ethernet VLAN Patch ID 112119-04 or subsequent compatible patch for supported versions of the Solaris 8 software. Sun Ethernet VLAN Patch ID 114600-02 or subsequent compatible patch supported versions of the Solaris 9 software.***

* The version number displayed from the `showplatform -v` command from the Sun Fire B1600 SC CLI printout refers to the BSC firmware version. The application software version is observed using the console `show version` command.

**Verify that you are using the supported Linux server module (xxx1.41-1) for the OS version on your Linux server.

***The patch currently installed can be displayed by entering `/usr/ccs/bin/mcs -p /platform/sun4u/kernel/drv/ce`. You can download patches from <http://sunsolve.sun.com>

The supported operating systems for the Sun Fire B10n blade are listed in TABLE 1-2.

TABLE 1-2 Supported Servers and Operating Systems

Operating System	Version	SPARC/x86	SF B1600 Server Blades			External Server		VLANs
			SF B100s	SF B100x	SF B200x	SF V60/65x	SF V210/240	
Solaris	8 HW 12/02	SPARC	X				X	Yes
Solaris	8 HW 5/03	SPARC	X				X	Yes
Solaris	8 HW 7/03	SPARC	X				X	Yes
Solaris	8 HW 2/04	SPARC	X				X	Yes
Solaris	9 8/03	SPARC	X				X	Yes
Solaris	9 12/03	SPARC	X				X	Yes
Solaris	9 4/04	SPARC	X				X	Yes
Red Hat Enterprise Linux (RHEL)	AS 2.1	x86		X	X			No
Red Hat Enterprise Linux (RHEL)	AS 2.1 Update-2	x86		X	X	X		No

TABLE 1-2 Supported Servers and Operating Systems

Operating System	Version	SPARC/x86	SF B1600 Server Blades			External Server		VLANs
			SF B100s	SF B100x	SF B200x	SF V60/65x	SF V210/240	
Red Hat Enterprise Linux (RHEL)	AS 3.0	x86		X	X	X		Yes
SuSe	SLES 8.0 SP-3	x86		X	X	X		Yes
x86 Solaris	9 8/03	x86		X	X	X		No
x86 Solaris	9 12/03	x86		X	X	X		No
x86 Solaris	9 4/04	x86		X	X	X		No

Zip files are available at the Sun Download Center at the following URL:
<http://www.sun.com/software/download/network.html>

Enter **B10n** and select Search the Download Center. Select Sun Fire B10n Content Load Balancing. Zip files listed in TABLE 1-3 are available.

TABLE 1-3 Zip Files Available at the Sun Download Center

Software Components	Zip File Name
BSC firmware	SunFire_B10n-version#-BSCFirmware.zip
B10n application software	SunFire_B10n-version#-Application.zip
Blade server module	SunFire_B10n-version#-SolarisModule.zip SunFire_B10n-version#-LinuxModule.zip (see TABLE 1-2 for the supported OS matrix)
Administration Guide and Product Notes	SunFire_B10n-version#-Docs.zip

Product Features

Key Features

- Two full-duplex Gigabit Ethernet interfaces
- Content load balancing based on URL, Cookies, or CGI scripts
- Server to client direct response
- Persistence

- 500 Layer 4 through Layer 7 rules
- Server, path, and blade failover
- Integrated management with the Sun Fire B1600 blade system chassis, the Sun blade servers, and SSL proxy blades
- Application monitoring through scripted health checking
- Round robin, weighted round robin, least connections, response time and static load balancing schemes
- External server load balancing

Server Selection Algorithms

- Round robin
- Weighted round robin
- Least connections
- Static

Supported Protocols

The Sun Fire B10n blade uses the following protocols for its services or management functions:

- TCP
- UDP
- HTTP
- HTTPS
- FTP
- TFTP
- ICMP
- ARP
- DNS
- Telnet

Diagnostic Support

- User executable self-test
- Power on self-test (POST)
- Manually invoked tests
- Error logging routines
- Debugging commands

Weighted Least Connections Feature

The load balancer monitors the servers using the CLB heartbeat message and the server module in the heartbeat response sets of the number of current open connections. This value is used by the load balancer to make the load balancing decision.

The underlying load balancing scheme in this mechanism is weighted round robin. The active connection count of the servers is used to compute the weight of each server dynamically and balance load based on the computed weight.

The number of connections active on a server indicates the amount of load a server is handling. Using this indication to dynamically adjust the weight of the servers maintains a steady load and minimizes the chances a server getting overloaded and others underloaded.

Every load balancing group could have a disparate set of servers with single and multiple CPU systems and various CPU frequencies in them, and could handle different connection rates. Doing a pure least connection algorithm in this environment could cause the less powerful (slower and fewer CPUs) servers to be overloaded while the more powerful system could be underloaded. To solve this problem each server is assigned a weighting factor. This factor in the CLI would be the same parameter as the weight specified in weighted round robin. The more powerful systems are given a higher weight and the slower systems are given a lower weight. When the relative connection count of the servers is computed, this weight is applied to the connections and then the new weight for the servers is computed.

The dependencies of the weighted least connections feature are as follows:

- Server Monitoring Frequency

Since the connection count is obtained from the server monitoring information in the heartbeat message, this scheme depends on the server monitoring frequency being fairly high. If the frequency is too low the active connections sample might not represent the current state of the system.

- Weight Update Frequency

The rate at which the weight is updated also affects the behavior. If the frequency is greater than the server monitoring frequency, then the updates will use the old connection count and is not of much use. If the frequency is too low, the weight might not reflect the current state of the system. This is not user configurable and is set at either compile time or read using property files.

Testing Weighted Least Connections

Unlike weighted round robin or round robin, there is not a specific indication of when the servers are correctly load balanced. This is because the weights are dynamically computed.

One possible way to know if the servers are correctly load balanced is to have some servers common in two different load balancing groups and have the group's load balancing scheme set to weighted least connection and observe if an average of all the server loads are approximately the same. If different types of servers are used, the weight should be applied to the connection count which is dividing the connection count of a server by its defined weight. These counts should be approximately equal.

Application Monitoring Feature

Protocol specific requests are sent to each server in the service and when the response is received, it is verified for correctness. The overall response time is also measured for use with response time based load balancing.

For HTTP, first a TCP connection is established and then a GET request for a specified URL is sent. When the response is received, the response code is checked for 200 (OK). The overall response time is also measured for use with response time based load balancing.

If the response code is not 200, the connection times out and the application is marked as down, and connections are not load balanced to that server.

The Role of the Content Load Balancing Blade

The Sun Fire B10n blade is a component within a larger system ultimately delivering highly available network services to a client population over an IP-based network. This section describes the role of such a highly integrated content load balancer within the larger system.

The minimal set of components comprising the system encompasses:

- One or more Sun Fire B1600 blade system chassis
- One or more Sun Fire B10n blades
- One or two Switch and System Controller (SSC) units per system chassis

- One or more servers. Servers can be any mix of blade servers housed in B1600 blade system chassis and stand-alone Sun servers external to the chassis but connected to the same Ethernet broadcast domain (Layer 2 network)

Additionally, the system may have:

- One or more SSL proxy blades
- External distribution switches and routers extending one or more of the networks
- Additional servers providing content, local name and configuration services, and aggregate management for one or multiple shelves. These servers may participate in the overall system by supporting various TFTP, NFS, DHCP, DNS, and N1 deployment related functions

In general terms, the intra-shelf network topology formed by connecting the Sun Fire B1600 system components is either a single or a dual redundant Layer 2 topology with blades “one-arm” connected to each of the switch fabrics. The switch fabric is VLAN partitionable for strict traffic isolation. SSC switches and uplinks can be used for a simple inter-shelf network, or connected to external distribution switches for larger configurations.

Note – This section defines the generic features and functions of the Sun Fire B10n blade. For more information about a specific firmware release, refer to the *Sun Fire B10n Content Load Balancing Blade Release Notes*.

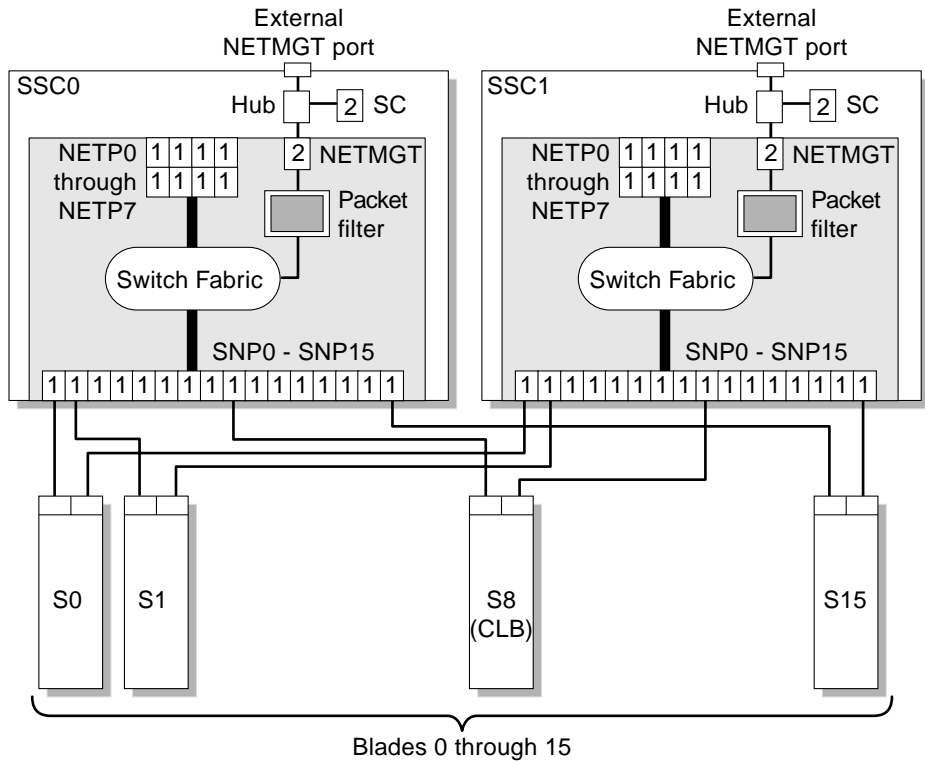


FIGURE 1-1 Ethernet Ports and Interfaces on the B1600 System Chassis and their Default VLAN Numbers

FIGURE 1-1 shows the intra-shelf network, where a Sun Fire B10n blade (shown in slot S8) can reside in any slot (S0 through S15) and connect to both SSC0 and SSC1 switch fabrics. The uplinks are labeled NETP0 through NETP7.

The numerals associated with each port (either 1 or 2), represent the VLAN numbers programmed into the system by default. The numbers indicate that there is one data VLAN (1), and one management VLAN (2). Further VLAN partitioning might be desirable as shown in FIGURE 1-2. The actual VLAN-ID assignment can be coordinated with the VLANs used in the external switches, or its scope can be limited to the internal switches, by keeping the uplinks as untagged VLANs.

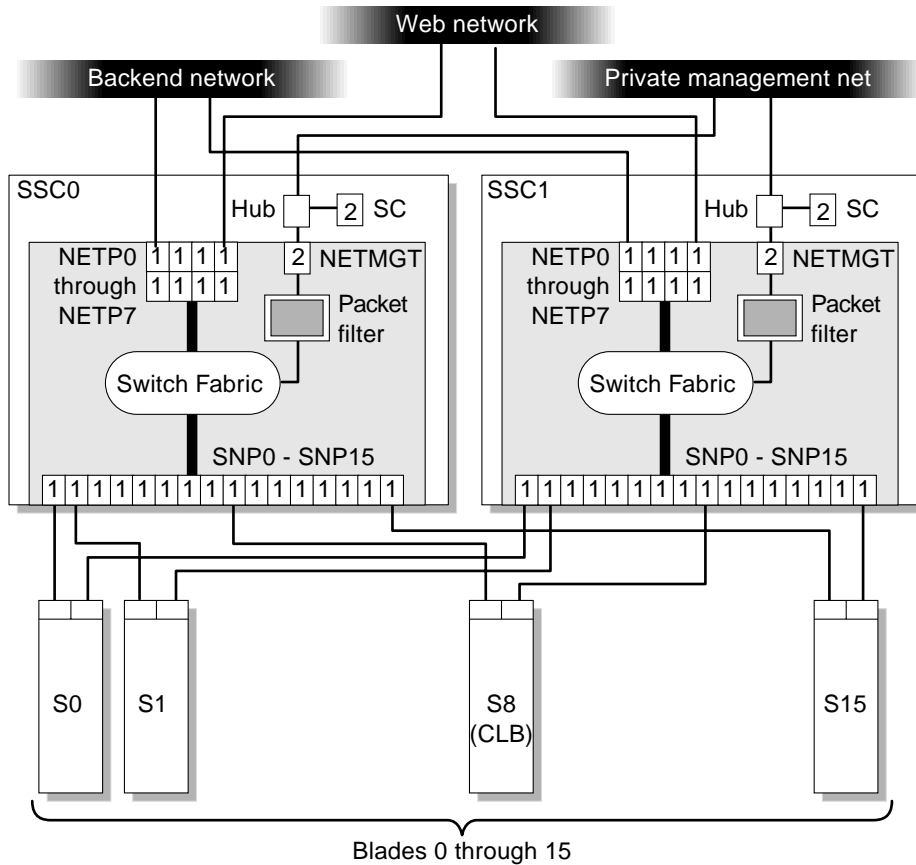


FIGURE 1-2 Dedicated Management Network and Web Server Network Isolated from the Backend Network

Load Balancing

The role of the content load balancer is to present a set of highly available network services. These services can be transported over http, TCP, or UDP, and are addressable through one or more Virtual IP addresses (VIPs), that the content load balancer is responsible for:

- Providing one level of address indirection so that the number and nature of actual servers can transparently evolve over time.
- Dividing requests among servers grouped in load balancing groups so that the total service demand can be satisfied through horizontal scaling.

- Maintaining persistence for clients or groups of clients requesting services that require affinity, that is, services where multiple consecutive requests must be satisfied by the same server.
- Delivering highly available services by taking responsibility for the failover functions that alter network paths, servers, and load balancer pairs upon service failure detection.
- Associating services to VLANs that partitions are based on meaningful criteria (service owner, back-end network, and the like).
- Participating one or more SSL proxy blades in the request packet flow whenever SSL encryption/decryption is necessary.
- Providing application monitoring through a user scripting interface.

VIPs are the routable IP addresses that clients obtain for the service through DNS lookups. A VIP address is *owned* by one content load balancer at a given time. VIPs are preserved through the content load balancer all the way to servers. Requests are directed to servers by rewriting their MAC addresses and their VLAN tags.

A service is identified by a 3-tuple comprising of the VIP, the Layer 4 protocol value (TCP or UDP), and TCP/UDP destination port. A multi-homed service can be associated with more than one 3-tuple. For example, two different VIPs can point to the same service. One of the initial steps in setting up the load balancer is to configure a service through the command line interface. Configurations can be input through the console interface by manual command line entry, scripted command sets or importing `config` files.

Load balancing or the server forwarding decision for a particular service is controlled by one or more rules. There are three basic types of rules, IP rules (layer 4), static HTTP and dynamic HTTP. An IP rule consists of source IP and source port, both have associated masks which enable subnets or port groups to be defined. Static HTTP (Layer 7) provides direct URL matching whereas dynamic HTTP provides wild carding of various parameters. Rule matching is based on a priority match where priority is determined by rule type or assigned priority. If IP rule priority is Low, the order is HTTP static, HTTP dynamic, and IP rule. If IP rule priority is set to High, the order is IP rule, HTTP static and HTTP dynamic. The number of bits configured determines order within a rule type.

The next parameter in the server decision is the type of load balancing scheme. The schemes currently supported are round robin, weighted round robin, least connections, response time and static.

A service, load balancing scheme and servers are associated by creating a load balancing group using the `lb-group` command.

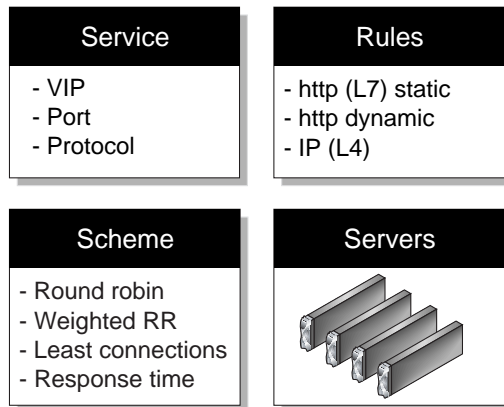


FIGURE 1-3 Load Balancing Group Configuration

For more details on command syntax for creating services, rules, and load balancing groups see Appendix D. Additionally, for an example, see Appendix B.

Topology Fundamentals

To match the ample switching capacity of the SSC units in the Sun Fire B1600 blade platform the content load balancer solution is designed to direct server responses toward clients without passing through the content load balancer. This enables the outbound capacity of the system to scale in proportion to the number of servers deployed, and to exploit the natural web traffic asymmetry where most of the traffic is server outbound.

To combine the uncompromised Layer 7 service performance with the direct server response, the content load balancing blade relies on a software module in each server. This server module contributes to the solution's high degree of integration by providing other key attributes, for example, path failover functionality.

The Sun Fire B1600 blade platform switches are separate networks, leaving the system designer the option to connect them externally and create a symmetrically configured redundant system where every blade is dual-homed, or to leave the switches segregated for a system where full redundancy is either not necessary (or achieved elsewhere in the system hierarchy), and blades are single-homed to separate networks. You can also create intermediate configurations where critical blades (content load balancers, proxies, and so on) reside on shelves with dual switches, but blade servers do not.

When you connect SSC switches to create redundant paths, it is best if:

- The interconnection occurs at the highest point in the network hierarchy

- The internal fabric of one shelf is connected directly to the corresponding fabric of another shelf (that is, daisy chain SSC0 with SSC0 and SSC1 with SSC1, and connect these uplinks at an external distribution switch, if any).

The above connections help ensure that the SSC switches are indeed leaf switches within the network infrastructure, and enable the content load balancer to use the shortest path within the redundant fabric (that is, the path that involves only one fabric).

FIGURE 1-2 illustrates nine shelves connected using a combination of distribution switches and internal SSC switches. Note that the SSC0 versus SSC1 fabric correspondence is preserved throughout the Layer 2 network, and that the fabrics are interconnected at the distribution switch level. In asymmetrical (capacity and hops) topologies like the one shown, it is also appropriate to house the content load balancing blades in shelves directly connected to the distribution switches.

Routers are shown for completeness as they represent the boundary of the Layer 2 network on the path towards the service clients.

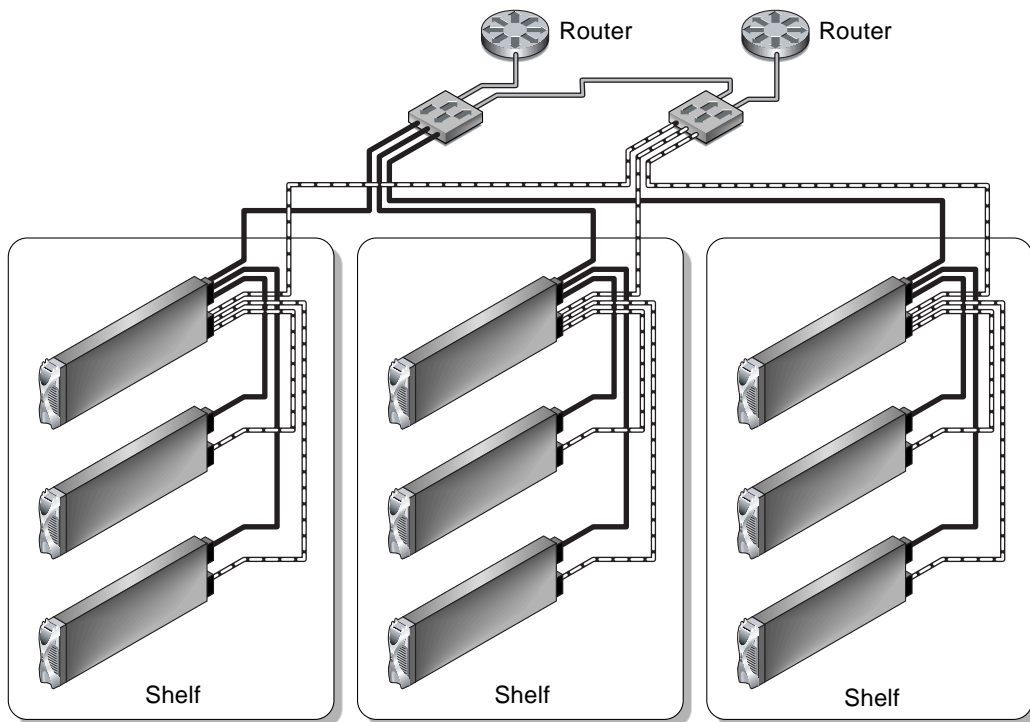


FIGURE 1-4 Sample Topology: Dual Tree Using External and Internal Switches

The Role of the SSL Proxy Blade

The SSL proxy blade is a companion product to the Sun Fire B10n blade, and as such its role is briefly described in this section. SSL proxy blades are used to:

- Consolidate and secure storage of server side certificates (servers remain stateless in terms of long-term secrets, and can be dynamically re-purposed or replaced).
- Accelerate RSA transactions and bulk encryption and decryption.
- Enable the content load balancer to perform Layer 7 load balancing on cleartext (decrypted) cookies and URLs.

For every service the content load balancer can be configured with one or more SSL proxy blades supporting the SSL sessions of the given service. SSL requests are delegated by the content load balancer and processed after decryption. Outbound encryption requests are directed from the servers to the corresponding SSL proxy blade without going through the content load balancing blade.

The content load balancing blade, along with its server-side module, are responsible for the appropriate path selection, failover, and VLAN tag selection for SSL traffic. The resulting functionality is summarized by:

- Secure traffic in cleartext form (after decryption or before encryption) is contained to a VLAN
- Secure traffic in cleartext form (after decryption or before encryption) is contained to a single fabric (and to a single shelf if all participating blades are in the same shelf)

This administration guide covers the configuration of SSL services at the content load balancing blade. Refer to the *Sun Fire SSL Proxy Blade Administration Guide* for more information.

Failover Alternatives

The service availability obtainable from a given system is a function of the intrinsic failure rates of its components and the automatic failover capabilities of the system itself. A system designed around Sun Fire B1600 blade system product family has the following service failover aspects:

1. Server failover – This is the ability of any load balancer to remove non-responsive servers from service groups so that new requests go to functional servers. This capability is based on the server-monitoring function.
2. Path failover – This is the ability of the system to use an alternate network path whenever the current path does not appear to work, because of cable, switch, link, or end-point faults. This type of failover tends to be transparent, in the sense that session state at all endpoints is still valid and usable.

3. Blade failover – This is the ability to deploy content load balancing blades in high availability (HA) pairs that monitor each other. For a given service one of the load balancing blades is a standby blade, identically configured to the active blade, and responsible for taking over in if the active blade fails. SSL devices do not monitor each other, and their failover is rather controlled by the load balancing blade monitoring them as if they were servers.

The system designer can decide which level of failover to design into the system:

- Server failover: always provided
- Path failover: possible whenever dual redundant switches are used. Controlled by the content load balancing blade, by its server module, and by configuring path failover on each server for server outbound path failover (towards the router).
- Blade failover: possible whenever content load balancing blades are deployed in pairs.

The Role of VLANs

The use of VLANs within the Sun Fire B1600 blade platform provides the ability to separate traffic into logical groups on the same Ethernet switch fabric. The use of VLANs is preferred for separation between data and management traffic and when using the Sun Fire B10p SSL proxy blade to create logical separation of client side traffic (between the specialty network blades and the switch uplinks) from the server side traffic (between the specialty network blades and the servers).

VLANs are configured at the SSC switches to create logical groups of endpoints that can communicate as if they were on the same LAN. VLANs also prevent or restrict traffic between endpoints on separate VLANs. However, some environments might not support VLANs and they may either not be configured or they may be disabled. For the SSL proxy blades, VLANs are set to on by default, but they can be disabled with the `set vlan filter disable` command from the CLI interface. Refer to the *Sun Fire B10p SSL Proxy Blade Administration Guide*.

SSL proxy blades are configured to enforce the separation and direction of client side versus server side VLANs. The content load balancer and the SSL proxy blade cooperate to enforce the association between the operation performed (encryption versus decryption) with the allowed direction to and from the client VLAN.

Switches are responsible for VLAN separation enforcement, based on the VLAN identifiers present on Ethernet packets (explicitly or implicitly), as well as the physical ingress and egress switch ports involved.

The scope of the VLANs may be confined to the Sun Fire B1600 shelves while keeping all the uplinks VLAN untagged, or alternatively tagging may be enabled in the SSC uplinks to extend VLANs through the external switch infrastructure; in this case the VLAN ID assignment must be consistent with the external switch/router infrastructure VLAN assignments.

The minimal set of VLANs recommended for a proxy blade system are:

- Client side VLAN
- Server side VLAN
- Management VLAN

TABLE 1-4 presents how the different VLAN assignments and the Sun Fire content load balancer and proxy blade duties accomplish the desired security outcome.

TABLE 1-4 VLAN Based Security

VLAN Involved	Requirement	Action
Management VLAN	<ul style="list-style-type: none"> • Confine the Sun Fire™ SSL proxy management to a management VLAN 	<ul style="list-style-type: none"> • Sun Fire SSL proxy only accepts management traffic on its management VLAN
Server side VLAN for secure traffic in its cleartext form	<ul style="list-style-type: none"> • Confine SSL traffic before encryption and after decryption to a VLAN 	<ul style="list-style-type: none"> • Server side VLANs configured at SSC for secure traffic includes just the relevant server(s), content load balancing blades, and SSL proxy blades. • Content load balancing blade responsible for transferring from client side VLAN to server side VLAN on ingress. • SSL proxy blade responsible for transferring from server side VLAN to client side VLAN on egress at encryption time. • Content load balancing module uses client side VLAN for cleartext egress traffic vs. server side VLAN for secure traffic in cleartext form.
Client side VLAN	<ul style="list-style-type: none"> • Prevent spoofing of traffic to be encrypted/decrypted 	<ul style="list-style-type: none"> • SSL proxy blade never encrypts/decrypts traffic arriving on client side VLAN.

The above actions, assigned to the different system components, must be complemented with the appropriate VLAN configuration at the SSC's and possibly other switches involved. The exact configuration scheme for the switches depends on how the uplinks are used and whether physical or VLAN separation is used.

VLANs can be extended to separate traffic for different user groups or tenants within the same Sun Fire B1600 blade platform.

In a multi-tenant environment it is appropriate to separate traffic based on the service 3-tuple. A VLAN identifier can be assigned by the content load balancing blade to identify the tenant (that is, the service owner), and thus ensure that its requests can only go to the specified tenant servers. In combination with the server side VLAN configuration, you can use VLANs to separate:

- Blade servers of different tenants
- Different tiers of a tenant (web tier, application server, NFS, and management)
- Pre- and post-encryption traffic of an SSL service

System Integration

Although this book describes the administration of the Sun Fire B10n blade at the lowest possible level, you may want to approach system integration (that is via CLI and scripting), it is certainly possible and desirable to achieve higher levels of integration abstraction with other Sun software products like N1 deployment, and Sun ONE Web and Portal Servers.

When considering system integration, the main Sun Fire B10n blade considerations are:

- Sun Fire B10n blade includes a specialized Layer 7 classifier engine, therefore there is no performance penalty for defining services at Layer 7 versus Layer 4.
- Layer 7 semantics may be used for either persistence (cookies) or content structure (URL), or both.
- Using Layer 7 semantics with HTTP 1.1 requires the content to be accessible from any of the servers in the server group, as HTTP load balancing occurs on TCP connection boundaries rather than request boundaries.

Usage Overview

The Sun Fire B10n blade has two levels of user access:

- Supervisor – In the Level 2 access mode, you can access all commands. You can add or change user name, password, and access level.
- General – In the Level 1 access mode, you can query the system status and configuration, but you cannot modify the configuration.

Command Modes

Some of the commands are accessible only with the right user access level. Even within a single user access level, there are different command modes.

- Config – In this mode, all the configuration commands are accessible. Only supervisor level users can access this mode by typing `config` at the command prompt.
- Non-config – Commands that do not affect the configuration are available in this mode.
- All – Listing commands (`show` and `dump`) can be accessed in any mode by any user.

The examples in this guide use the following format:

```
hostname (command_mode) {username}# command
```

For example:

```
puma {guest}# show user
```

- *hostname* is puma
- *command-mode* is non-config
- *username* is guest (general)
- *command* is show user

```
puma (config) {admin}# ip interface 0 192.50.50.134 mask
255.255.255.0
```

- *hostname* is puma
- *command-mode* is config
- *username* is admin (supervisor)
- *command* is ip interface 0 192.50.50.134 mask 255.255.255.0

Installing the Blade and Setting Up the System

This chapter describes how to install the hardware.

This chapter contains the following sections:

- “Installing the Hardware” on page 21
- “Location of Ports” on page 26
- “Serial Port Pin Numbers” on page 29
- “Powering On Content Load Balancing Blades” on page 30
- “Powering Off Content Load Balancing Blades” on page 32
- “Replacing Your Sun Fire B10n Blade” on page 35

Installing the Hardware

The instructions in this section are specific to installing the Sun Fire B10n blade into the Sun Fire B1600 blade system chassis. However, you should still refer to the documentation that came with your system for more complete information.

▼ To Install the Hardware

Note – To ensure adequate airflow, you must populate all 16 blade slots of the Sun Fire B1600 blade system chassis with either blades or filler panels before you apply power to the system chassis. Do not leave any slot empty.

- 1. Remove the filler panel from an unpopulated slot in a Sun Fire B1600 Blade system chassis or other supported system.**

Insert your finger in the pull recess located in lower portion of the filler panel lever and pull gently to disengage the locking mechanism (FIGURE 2-1).

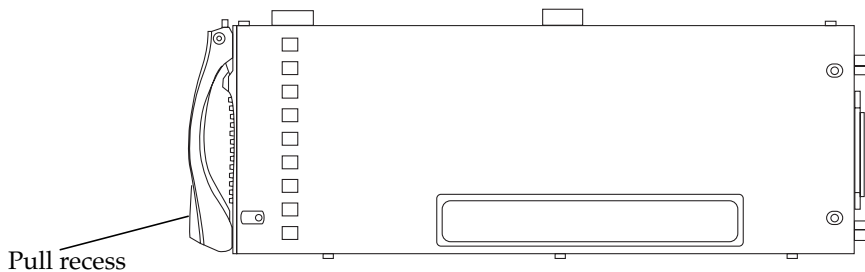


FIGURE 2-1 The Filler Panel Locking Mechanism



Caution – Pulling the blade lever will disengage it from the Sun Fire B1600 chassis and the blade will power off immediately. Please make sure that the blade is gracefully shutdown or powered off from the B1600 SC command line interface.

2. Pull the lever mechanism in a forward and upward motion, causing the filler panel lever to unlatch and eject the filler panel partially from the system chassis (FIGURE 2-2).

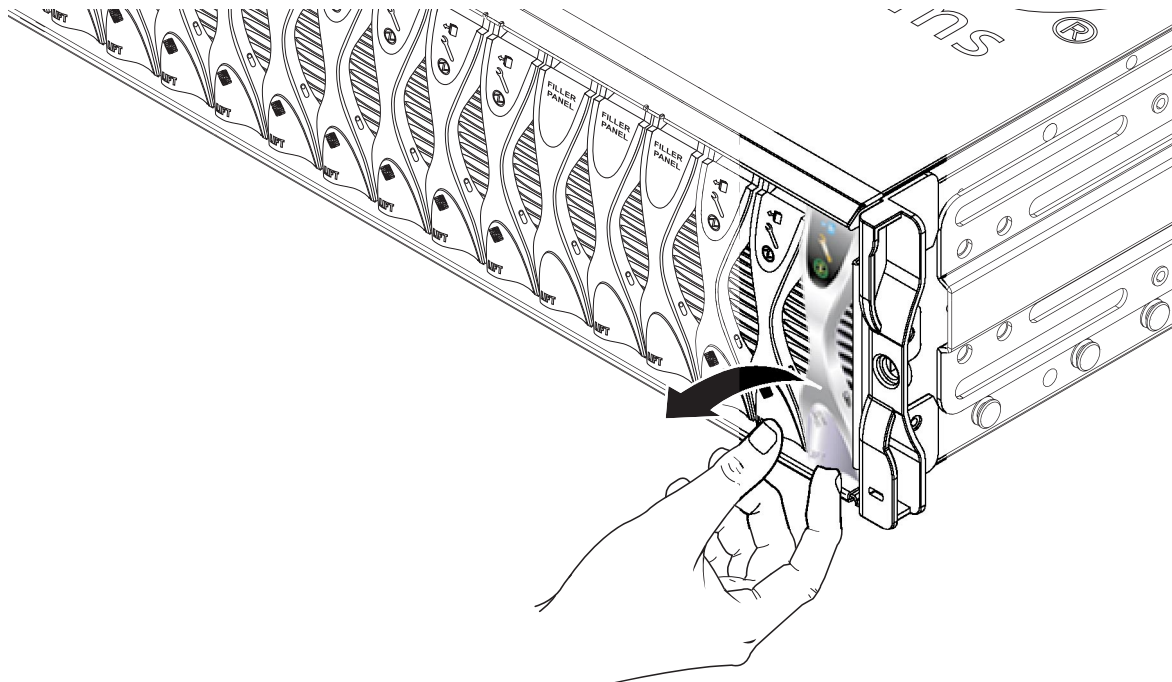


FIGURE 2-2 Disengaging the Blade-Locking Mechanism

3. Pull the lever to remove the filler panel from the system chassis (FIGURE 2-3).

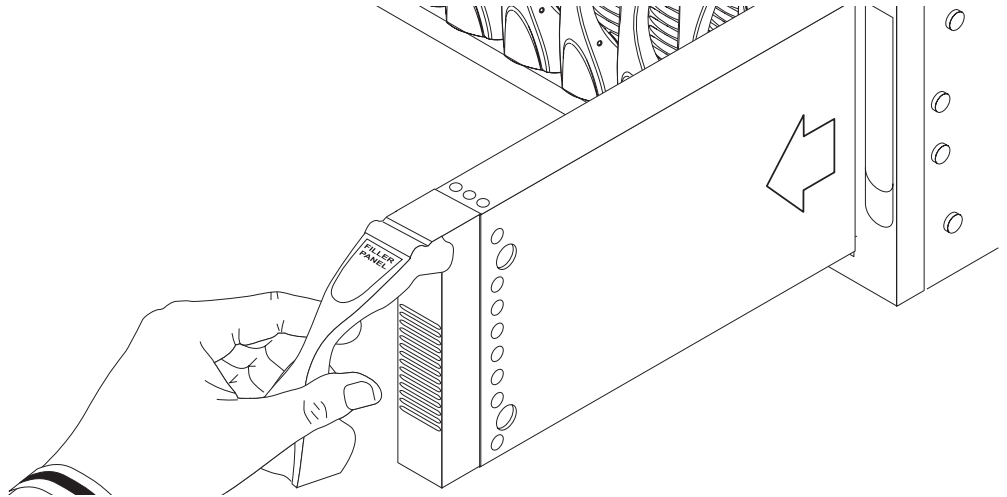


FIGURE 2-3 Removing the Filler Panel

4. If required, open the blade lever by inserting a finger in the pull recess located in lower portion of the blade lever and pull the lever mechanism in a forward and upward motion, causing the lever to unlatch (FIGURE 2-4).

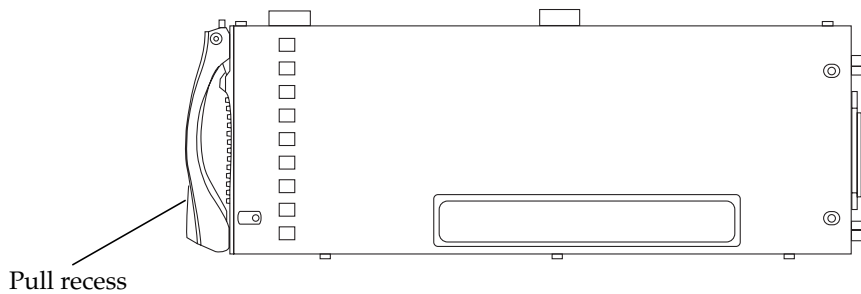


FIGURE 2-4 Blade Locking Mechanism

5. Align the Sun Fire B10n blade with an empty slot in the system chassis.

Ensure that the blade connector is facing towards the system chassis, with the hinge point of the lever mechanism uppermost, and support the bottom of the blade with your free hand while lifting the blade up to the system chassis (FIGURE 2-5).

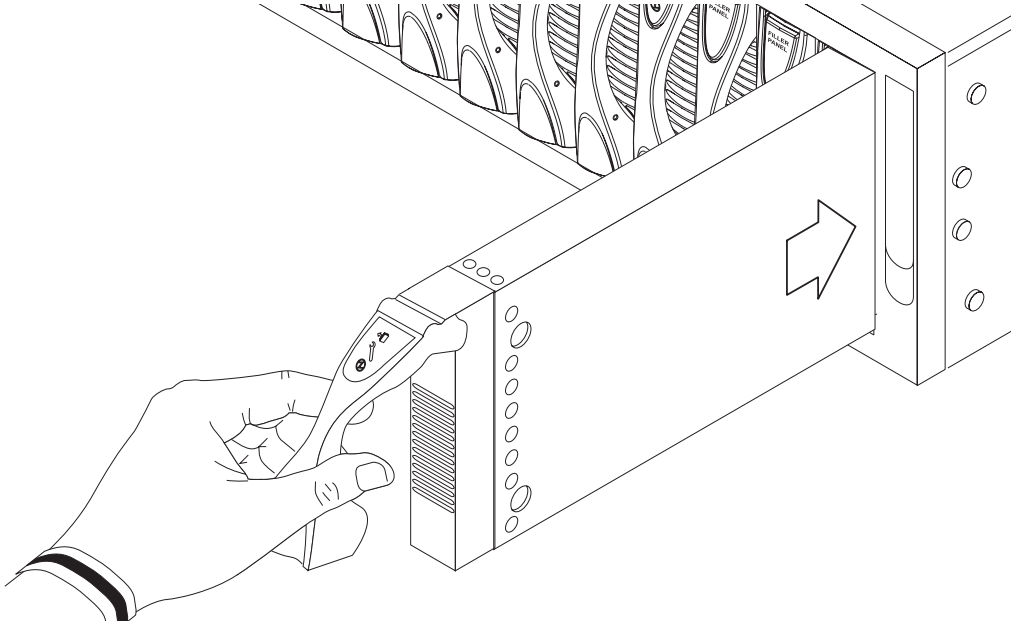


FIGURE 2-5 Aligning and Inserting the Blade

6. Insert the Sun Fire B10n blade into the system.

Caution – Ensure that the blade engages with the system chassis guidance system. Failure to align the blade correctly can result in damage to the chassis midplane or the blade connection.

- 7. Gently push the blade into the slot until the blade latch ears, on top of the lever, are positioned in the chassis (FIGURE 2-6).**
- 8. Complete the hardware installation by closing the blade lever fully, which engages the blade into the chassis slot (FIGURE 2-6).**

The green LED flashes as the blade powers up, and glows steadily when the blade is up and running.

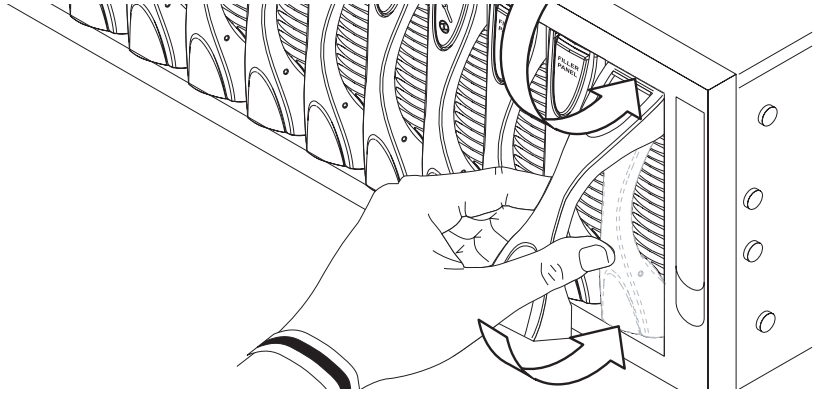


FIGURE 2-6 Closing the Blade Lever Mechanism

LED Displays

Use the LEDs on the individual system components to determine if the system is operating normally. Monitor LEDs routinely on the:




- Power Supplies
- Load balancing blades

The LEDs can be off, on, or flashing. When the fault LED is on (lit), this indicates that a fault has occurred in the component. A fault is any condition that is considered to be unacceptable for normal operation. When the fault LED is lit, you must take immediate action to clear the fault. You can only remove a hot-swappable component when the blue Removal OK LED is lit.

TABLE 2-1 lists the LED status codes for the following hot-swappable components:

- Blades
- Power Supply

TABLE 2-1 Blade and Power Supply Status Codes

Power (Green)	Fault (Amber)	OK to Remove (Blue)	Indication	Corrective Action
				
Off	Off	Off	Component not operating. Fault condition unknown.	You can remove a component from the system.
Off	On	Off	Component not operating. Fault condition present.	You cannot remove a component from the system.
Off	Off	On	Component not operating. No fault condition present.	You can remove a component from the system.
Off	On	On	Component not operating. Fault condition unknown.	You can remove a component from the system.
On	Off	Off	Normal component operation.	N/A
On	Off	On	Component not operating. No fault condition present.	You can remove a component from the system.
On	On	Off	Component operating. Fault condition present.	You cannot remove a component from the system.
On	On	On	Component operating. Fault condition present.	You can remove a component from the system.
Flashing	Off	Off	Component is powering up.	N/A

Location of Ports

All ports and are located at the back of the Sun Fire B1600 blade system chassis. These connections are shown in FIGURE 2-7.

Note the location of the following ports:

- 10/100BASE-T network management cables
- 10/100/1000BASE-T data network cables
- RS232 serial cables

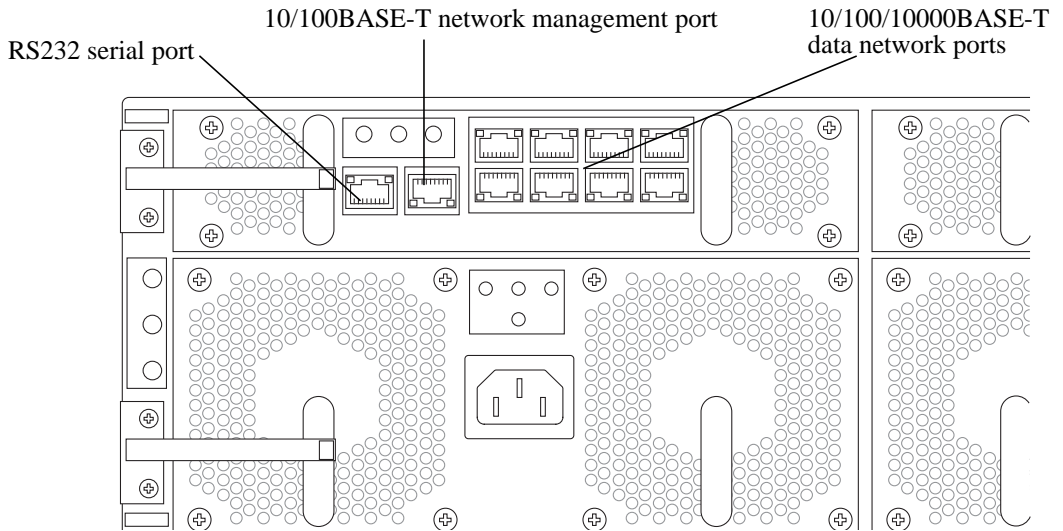


FIGURE 2-7 External Cable Connections



Caution – Do not connect a telephone jack connector to any RJ-45 port. This can damage the switch. Instead, use only twisted-pair cables with RJ-45 connectors that conform with FCC standards, or local national wiring or electrical regulations.

Note – Twisted-pair cables must not exceed 328 feet (100 meters).

Connecting to the 10/100/1000BASE-T Data Network Ports

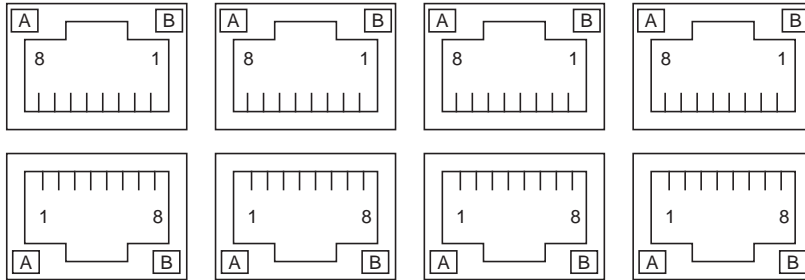


FIGURE 2-8 The 10/100/1000BASE-T Data Network Ports

Arranged as a 4x2 array, these RJ-45 ports provide the connection to the combined switch and service processor. Each port has integral green Link Present and Link Active LED indicators.

Note – The Link Present indicator is always on the left, regardless of the orientation of the RJ-45 port.

TABLE 2-2 10/100/1000BASE-T Data Network Port Pinouts

Pin 1	Pin 2	TRD0-
Pin 3	Pin 4	TRD2+
Pin 5	Pin 6	TRD1-
Pin 7	Pin 8	TRD3-
LED A	LED B	Link Active

Serial Port Pin Numbers

Viewing the Sun Fire B1600 blade system chassis from the back, pin 1 of the RJ-45 serial port is on the left, and pin 8 is on the right.

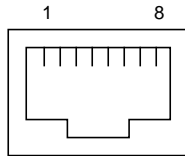


FIGURE 2-9 Serial Port Pin Numbers

TABLE 2-3 Serial Port Pinouts

Pin number on System Chassis	Signal
Pin 1	RTS
Pin 2	DTR
Pin 3	TXD
Pin 4	Signal Ground
Pin 5	Signal Ground
Pin 6	RXD
Pin 7	DSR
Pin 8	CTS

Powering On Content Load Balancing Blades

Note – To power on any content load balancing blade, you must have access to the system controller and r-level user permission. Refer to the *Sun Fire B1600 Blade System Chassis Administration Guide* (817-4765-11) for information on system controller user permissions.

- To power on a single blade, type:

```
sc> poweron Sn
```

Where *S* indicates the slot and *n* is the number of the slot containing the blade you want to power on. Valid slot numbers range from 0 to 15.

- To power on more than one blade, specify each blade in a space-separated list as in the following example:

```
sc> poweron S6 S11
```

Or, if any of the blades is in a continuous range, you can specify the range. For example, to power on the blades in slot 6 and also the blades in slots 8 through 10, you would type the following:

```
sc> poweron S6 S[8-10]
```

Note – The Sun Fire B1600 system controller can be configured to automatically power on the blades. Please refer to the *Sun Fire B1600 Blade System Chassis Administration Guide* (817-4765-11) for information on setting up this feature.

- Use the `showplatform` command to verify the status of the B10n blade:

```

sc> showplatform

FRU          Status          Type
-----
S0           OK              SF B100s
S1           OK              SF B100s
S2           OK              SF B100s
S3           OK              SF B100s
S4           Not Present    ***
S5           Not Present    ***
S6           Not Present    ***
S7           Not Present    ***
S8           Not Present    ***
S9           Not Present    ***
S10          Not Present    ***
S11          Not Present    ***
S12          Not Present    ***
S13          Not Present    ***
S14          OK              SF B10n
S15          OK              SF B10n
SSC0         OK              SF B1600 SSC
SSC0/SC
SSC0/SWT
SSC1         Not Present    ***
SSC1/SC
SSC1/SWT
PS0          OK              SF B1600 PSU
PS1          OK              SF B1600 PSU
CH           OK              SF B1600

Domain      Status
-----
S0          OS Running
S1          OS Running
S2          OS Running
S3          OS Running
S14         OS Running
S15         OS Running
SSC0/SWT   OS Running
SSC0/SC    OS Running (Active)

```

Note – Slots 14 and 15 show that the B10n blade is OK.

Powering Off Content Load Balancing Blades

Note – To power off any content load balancing blade, you must have access to the system controller and r-level user permission. Refer to the *Sun Fire B1600 Blade System Chassis Administration Guide* for information on system controller user permissions.

Note – All the various options in this section can be used on the same command line except for the `-r` and `-s` commands: these two are alternatives to each other.

Powering Off With an Orderly Shutdown of the Operating System

The `poweroff` command attempts to shut down the operating system on a blade or blades in an orderly fashion. The command also prompts you to confirm that you intend to shutdown the blade or blades you have specified.

- To power off a single blade, type:

```
sc> poweroff Sn
```

Where *n* is the number of the slot containing the blade you want to power off.

Forcing the Power Off

The `poweroff` command attempts to shutdown the operating system on a blade in an orderly fashion. If this orderly shutdown fails on a particular blade, the `poweroff` command will not continue to power off the blade.

- To force the blade to power off even if an orderly shutdown has failed, include the `-f` option on the command line, as in the following example:

```
sc> poweroff -f Sn
```

Where *n* is the number of the slot containing the blade you want to power off.

Powering Off a Load Balancing Blade Without Requiring the Confirmation Prompt

When you run the `poweroff` command to power off a blade, you are prompted to confirm that you intend to power off the blade you have specified.

- To avoid receiving the confirmation prompt when you use the `poweroff` command, include the `-y` option on the command line.

For example:

```
sc> poweroff -y Sn
```

Where *n* is the number of the slot containing the blade you want to power off.

Powering a Load Balancing Blade Down to Standby Mode

There are two ways to power a blade down to standby mode. You can use either the `standbyfru` command or the `poweroff` command.

- To power down a blade or blades to standby mode using the `poweroff` command, type:

```
sc> poweroff -s Sn
Are you sure you want to power off FRU S5 (y/n)?: y
S5: Poweroff sequence started.
Sep 19 23:09:05: MINOR: S5: Powered off.
Sep 19 23:09:05: MINOR: S5: Active LED state changed to SLOW FLASHING
```

Where *n* is the number of the slot containing the blade you want to power down. When a blade is in standby mode, the system service processor continues to monitor its operational state.

Note – You cannot use the `-s` option on the same command line as the `-r` option.

- To power down a blade to standby mode using the `standbyfru` command, type:

```
sc> standbyfru Sn
```

Where *n* is the number of the slot containing the blade you want to power down. When a blade is in standby mode, the system service processor will continue to monitor its operational state.

- To power down more than one blade to standby mode, specify each blade in a space-separated list, as in the following example:

```
sc> standbyfru S6 S11
```

Powering Off a Content Load Balancing Blade to Remove It

- To power down a blade or blades for removal, type:

```
sc> poweroff -r Sn
```

Where *n* is the number of the slot containing the blade you want to power down. When a blade is powered off for removal, the OK to Remove LED is lit.

Note – You cannot use the `-s` option on the same command line as the `-r` option.

Replacing Your Sun Fire B10n Blade

The upgraded Sun Fire B10n blade has the following features:

1. The 1.0 BSC firmware
2. Two B10n boot images—version 1.0.1 and 1.1. The default boot image is 1.1.
3. The B10n bootrom, version 1.1.

▼ To Export the Configuration From the Old Board

1. Go to the `/RFA0` directory.

```
puma{admin}# cd /RFA0
```

2. Archive the `CONFIG` directory:

```
puma{admin}# tar lbconfig.tar CONFIG
```

3. Export the configuration archive file:

```
puma{admin}# export lbconfig.tar
The FTP server address: <ftp_server_ip>
The source directory path: type [cr] to use current directory:
  (null) source path, using current directory
The source file name: lbconfig.tar
The destination directory path: <path_on_ftp_server>
The destination file name: lbconfig.tar
The user name: <user_name_for_ftp_server>
The user password: <user_password_for_ftp_server>
export file succeed!
```

▼ To Import the Configuration to the Upgraded Board

1. Power off the old board and remove it from the chassis.

2. Install the upgraded board.

The board comes up with an empty configuration with the B10n 1.1 application image running.

3. Configure the network interface. Optionally, configure the management VLAN (if applicable).

4. Go to the `/RFA0` directory:

```
puma{admin}# cd /RFA0
```

5. Import the 1.0 or 1.1 configuration:

```
puma{admin}# import filename
    The FTP server address: <ftp_server_ip>
The source directory path: <path_on_ftp_server>
The source file name: lbconfig.tar
    The destination directory path:
    (null) path, using current directory...
    The destination file name: lbconfig.tar
    The user name: <user_name_for_ftp_server>
The user password: <user_password_for_ftp_server>

import file succeed!
```

6. Unarchive the configuration file.

```
puma{admin}# tar xvf lbconfig.tar
```

7. Reboot the B10n blade to get the imported configuration:

```
puma{admin}# reboot
```

Note – To run traffic with B10n 1.3.5 application image, the blade server module has to be updated to version 1.3.5.

Preparing the Sun Fire B10n Blade for Load Balancing

To prepare the Sun Fire B10n blade for load balancing, you must first configure the blade servers, then set up the content load balancing blade. This chapter describes the procedures for preparing the system for load balancing.

This chapter includes the following sections:

- “Configuring the Blade Servers” on page 37
- “Setting Up the Load Balancing Blade” on page 41
- “Completing the Basic Configuration” on page 44

Configuring the Blade Servers

The Sun Fire B10n software includes the following components:

- Blade server module
- B10n application software
- BSC firmware

See “Software Architecture” on page 2 to understand the different software components.

The procedures for downloading both the blade server module software and the B10n application software as well as the firmware on the Sun Fire B10n blade involve downloading the software to your TFTP server.

If you are updating the blade server module before downloading the software, check the blade server module software version.

▼ To Check the Blade Server Module Software Version

1. At the `sc` prompt enter the following command:

```
sc> console Sn
```

Where *S* indicates the slot and *n* is the number of the slot containing the blade you want to access. Valid slot numbers range from 0 to 15.

2. At the Solaris root prompt, check the module information:

```
# modinfo | grep clbmod
```

The following example checks the blade server module software of the blade in slot 10:

```
sc> console s10
Connected with input enabled on fru S10
Escape Sequence is '#.#.')
```

The information shows that software version 1.34 is currently installed.

```
# modinfo | grep clbmod
213 78160000 d18f 21 1 clbmod (Server CLB module v1.52)
```

▼ To Set Up a Solaris SPARC Blade Server Module

1. To download the latest software, go to the following site and select Sun Fire B10n:
<http://www.sun.com/software/download/network.html>
2. Unzip the Solaris SPARC server module version 1.2 zip file.

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-SolarisModule.zip
```

3. Install the Solaris SPARC blade server module software packages:

```
# cd path_to_unzipped_file/Solaris/sparc
# pkgadd -d .
```

A message similar the following is displayed:

```
1  SUNWclbut      Sun Content Load Balancing Utilities
                        (sparc) 1.3,REV=2004.02.12
2  SUNWclbx.u     Sun Content Load Balancing Module (64-bit)
                        (sparc.sun4u) 1.3,REV=2004.02.12

Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

4. Press return to install all the packages.

5. Install the blade server module on each blade server.

The blade server module (clbmod) resides in the /kernel/strmod/sparcv9 directory. The blade server module must be installed on each blade server individually. You can install the packages from the blade server to the TFTP server.

6. Configure the management IP address to the server interface:

```
# /usr/sbin/ifconfig ce0 plumb ip-addr netmask netmask up
```

7. To configure interfaces to remain across reboots, add the interfaces to `/etc/opt/SUNWclb/clb.conf` placing each interface on a separate line to be configured at start up. Or stop and start the `clbctl` script after creating the file and adding the interfaces.

```
# /etc/init.d/clbctl stop
# /etc/init.d/clbctl start
```

Following is a sample configuration file:

```
#
# Copyright (c) 2003 by Sun Microsystems, Inc.
# All rights reserved.
#
# ident "@(#)clb.conf 1.3 03/01/08 SMI."
#
# This file lists the interfaces to be configured with the content
# load balancing module (CLB). On boot these interfaces are
# configured for content load balancing. The interfaces are
# specified one per line.
# Interfaces configured for VLAN and are part of the load balancing
# are also
# listed here.
# Example:
cel23000
cel
ce0
```

8. Configure the virtual IP addresses (VIPs) that this server supports:

```
# ifconfig virtual loopback interface plumb vip netmask netmask up
```

You must configure the virtual IP address of the service being load balanced. If it is not configured, the stack will not recognize the destination address in the incoming IP packets and will reject packets received for that destination address. Configure the virtual IP address on the loop back interface as a logical IP address. Using the loop back interface prevents responses to incoming ARP request broadcasts for the virtual IP address. Use different loop back instances for different virtual IP addresses.

For example if the server supports VIP 192.50.50.1, you would type the following:

```
# ifconfig lo0:1 plumb 192.50.50.1 netmask 255.255.255.0 up
```

Setting Up the Load Balancing Blade

To make the content load balancing blade functional, you must first set it up. Once you are at the admin prompt, you can get help by doing either of the following:

- **At the admin prompt, type the following command:**

```
puma{admin}# help command_name
```

- **At the admin prompt, type a question mark after a command:**

```
puma{admin}# config service ?
```

▼ To Set Up a Content Load Balancing Blade

1. **Connect your Telnet console with a Sun Fire B1600 blade system chassis serial port.**
2. **Telnet into the system controller (sc):**

```
% telnet sc_ip-addr
```

Where *sc_ip-addr* is the IP address of the system controller.

3. **To power on a single content load balancing blade, type the following:**

```
sc> poweron Sn
```

Where *S* indicates the slot and *n* is the number of the slot containing the blade you want to power on.

If you are powering on more than one blade, list the slot number for each blade you are powering on. See “Powering On Content Load Balancing Blades” on page 30.

4. **Access the console for the blade:**

```
sc> console Sn
```

5. Login as `admin` to access the command line interface:

```
Login:admin
Password:admin
puma{admin}#
```

Note – The default login and password for the administrator is `admin`. To ensure the security of the configuration, change the default password before you continue. The new password must have at least six characters. See “User Access” on page 48 for more information.

6. Change the default password:

```
puma{admin}# password admin
Enter new password: new admin secret password
Confirm new password: new admin secret password
```

7. Enter config mode:

```
puma{admin}# config
puma(config){admin}#
```

8. Configure an IP interface:

```
puma(config){admin}# ip interface 0 ip-addr mask netmask
```

This command sets up the interface 0 (`iq0`) on your content load balancing blade, that is the interface connected to switch SSC0 (in slot 0).

Note – The switch number, for example SSC0, corresponds to the slot where the blade server resides.

The following example sets the IP address on interface 0 at 192.50.50.134 :

```
puma(config){admin}# ip interface 0 192.50.50.134 mask
255.255.255.0
```

Each Sun Fire B10n blade has two interfaces. Alternatively, you could configure the second interface as follows:

```
puma(config){admin}# ip interface 1 ip-addr mask netmask
```

This alternative configuration sets up the interface 1 (iql) on your content load balancing blade, that is the interface connected to switch SSC1 (in slot 1).

The following example sets the IP address on interface 1 at 192.50.50.135:

```
puma(config){admin}# ip interface 1 192.50.50.135 mask
255.255.255.0
```

9. Verify that the interface is working:

```
puma(config){admin}# ping remote-ip-addr
```

The following example pings the remote IP address at 192.50.50.200 :

```
puma(config){admin}# ping 192.50.50.200
```

Note – Ensure that the *remote-ip-addr* is reachable from the Sun Fire B10n blade before you complete the basic configuration.

Completing the Basic Configuration

Before the Sun Fire B10n blade can be configured to do basic load balancing, you must at minimum, configure a default gateway, and DNS server, a service, and a group.

▼ To Configure a Default Gateway

- As admin in config mode, set the default gateway:

```
puma(config){admin}# default gateway ip-addr
```

▼ To Configure the DNS Server

- As admin in config mode, configure the primary DNS server:

```
puma(config){admin}# dns server 192.50.50.100 primary
```

▼ To Configure DNS Suffix

- As admin in config mode, configure the DNS suffix:

```
puma(config){admin}# dns suffix mycompany.com
```

▼ To Commit the Configuration

- As admin in config mode, commit the configuration you just set up:

```
puma(config){admin}# commit
```


▼ To Verify the Configuration

- As admin, verify the configuration you just set up:

```
puma{admin}# show network
```

```
Default Gateway           : 192.50.50.200
Hostname                  : puma
DNS Primary               : 192.50.50.100
DNS Secondary             : Not Configured
DNS Suffix                : mycompany.com
Server monitor interval  : 3
Server monitor max-try   : 5
Path Failover Status     : Enabled
Path Failover Target on interface 0 : 192.168.101.81 (Path Up)
Path Failover Target on interface 1 : 192.168.101.82 (Path Down)
Path Failover monitor interval : 5
Path Failover monitor max-try : 5
```

```
Network Interface Table:
```

```
=====
If      IP Address      Mask           MAC Address      Status  Link
-----
0       192.168.101.251 255.255.255.0 00:03:ba:2c:73:a0 Up      Up
1       0.0.0.0          0.0.0.0        00:03:ba:2c:73:a1 Down    n/a
=====
```

```
System VLAN Table:
```

```
=====
VLAN Type                VLAN ID      Status
-----
Management                18          Enabled
Data                       28          Enabled
=====
```

Note – The output from the `show network` command is only an example of the data provided. Your output will be different.

Command-Line Options

This chapter describes the management and control interfaces available through the Sun Fire B10n blade command line interface (CLI). It lists the CLI commands under the various management categories with the appropriate options.

This chapter includes the following sections:

- “Typographic Conventions Specific to the Sun Fire B10n Command Line Interface” on page 48
- “User Access” on page 48
- “Configuring the Networking” on page 52
- “Monitoring” on page 58
- “Configuring SSL Device Entries” on page 61
- “Configuring Multiple SSL Devices” on page 67
- “Configuring the Content Load Balancing Blade” on page 67
- “Load Balancing Service Configuration and Management” on page 70
- “Server Configuration” on page 91
- “Load Balancing Rule Configuration” on page 93
- “Load Balancing Group Configuration and Management” on page 98
- “Load Balancing Configuration Listings” on page 109
- “Configuring the System” on page 112
- “Flash File System Commands” on page 127
- “Other Useful Commands” on page 130

Note – While you can enter multiple commands in the CLI, no command can be longer than 255 characters. Hence, commands longer than that must be divided into as many commands as necessary to stay within the 255 character limitation.

Typographic Conventions Specific to the Sun Fire B10n Command Line Interface

Command descriptions use these conventions:

- Commands and keywords are in **boldface** in code boxes, and in *courier* in tables and text.
- Arguments for which you supply values are in *italic* in code boxes, tables, and text.
- Required options are grouped in braces ({ }) and separated by vertical bars (|).
- Optional elements are grouped in square brackets ([]) and separated by vertical bars (|).

Note – Keywords are not case-sensitive, but user-specified values are.

User Access

The Sun Fire B10n blade has two levels of user access:

- Supervisor – In the Level 2 access mode, you can access all commands. You can add or change user name, password, and access level.
- General – In the Level 1 access mode, you can query the system status and configuration, but you cannot modify the configuration.

TABLE 4-1 describes the user commands.

TABLE 4-1 user Commands

Command	Description
<code>user password</code>	Changes the password. System responds with a prompt for the new password. Then prompts for confirmation of the new password.
<code>user access</code>	Access level assigned to a user. 1 = the general access level with read-only permission. 2 = the highest access level with access to all commands.

TABLE 4-1 user Commands (Continued)

Command	Description
user add	Adds a user.
user delete	Deletes a particular user's ability to access the system.
user show show user	Lists all users currently existing in the system, along with their respective access levels.

Using the login Command

The `login` command is used to log in initially as the administrator. It can also be used to log in as another user with a different access level than the one you currently have.

The `login` command has no defaults and can be used at any access level. The corresponding command is `logout`.

Note – For security reasons, always `logout` before you leave the console.

▼ To Log In as Administrator

1. Access the console for the blade:

```
sc> console Sn
```

Where `S` indicates the slot and `n` is the number of the slot containing the blade you you want to configure.

2. Log in as `admin` to access the command-line interface:

```
Login: admin  
Password: admin  
puma{admin}#
```

Note – The default `admin` password is `admin`. To ensure security, change the default password before configuring the content load balancing blade.

3. Change the admin password:

```
puma{admin}# user password admin  
Enter new password:  
Confirm new password:
```

Adding Users

Only the administrator (Supervisor) with Level 2 access can add new users who can be given all the privileges of the administrator or limited privileges, depending on the assigned access level. By default, the user is created with Level 1 access.

Note – If you do not specify the access level when you add a user, the default access (Level 1) will be used.

TABLE 4-2 lists the parameters for `user access` command:

TABLE 4-2 Parameter Description for User Access

Parameter	Description
<i>username</i>	Login name of the user.
<i>access</i>	Qualifier for the access level assigned to the new user.
1	General (intermediate) access level.
2	Supervisor (highest) access level.

▼ To Add a User

- As admin, you can add new users, assign access level, and a default password:

```
puma{admin}# user add name username [access {1|2}]  
puma{admin}# user access name username access {1|2}  
puma{admin}# user password login-name
```

▼ To Change the User Access Level

- As admin, you can change a user's access level:

```
puma{admin}# user access name username access {1|2}
```

The following example changes the access level of user1 to 1.

```
puma{admin}# user access name user1 access 1
```

▼ To Change the User Password

- As admin, you can change a user's password level:

```
puma{admin}# user password username  
Enter new password:*****  
Confirm new password:*****  
puma{admin}#
```

▼ To Remove a User

- As admin, you can remove a user:

```
puma{admin}# user delete username
```

▼ To List All Users

- As admin, you can list all users:

```
puma{admin}# user show
```

You can also list all users with the `show user` command:

- **As admin, you can list all users:**

```
puma{admin}# show user
```

Both commands list all users currently existing in the system, along with their respective access levels.

Configuring the Networking

This section describes how to configure the network for the Sun Fire B10n blade

▼ To Configure the Management IP Address

1. **As admin, use the following command:**

```
puma{admin} # config
```

2. **As admin in config mode, use the following command:**

```
puma(config){admin}# ip interface {0|1} ip-addr mask netmask
```

The `config ip interface` command configures the IP address on the content load balancing blade to be used for management and control. Use this IP address for tasks such as opening a Telnet session on the content load balancing blade.

TABLE 4-3 describes the parameters for setting the real IP addresses.

TABLE 4-3 Parameters for Setting the IP Address

Parameter	Description
<code>interface</code>	Qualifier for the interface
0	Interface 0
1	Interface 1

TABLE 4-3 Parameters for Setting the IP Address (*Continued*)

Parameter	Description
<i>ip-addr</i>	IP address.
<i>mask</i>	Qualifier for the subnet mask.
<i>subnet</i>	Subnet mask for the real address.

Examples

The first example sets the IP address on interface 0 as 192.50.50.144 and the subnet mask as 255.255.255.0.

```
puma(config){admin}# ip interface 0 192.50.50.144 mask 255.255.255.0
```

The following example sets the IP address on interface 1 as 192.50.50.145 and the subnet mask as 255.255.255.0.

```
puma(config){admin}# ip interface 1 192.50.50.145 mask 255.255.255.0
```

Note – The IP addresses shown in Examples 1 and 2 are different from virtual IP (VIP) addresses. The Sun Fire B10n blade does not load balance traffic destined to these IP addresses.

▼ To Send a ping Request

- As any user, use the following command:

```
puma{user} # ping {ip-addr | hostname} [packet-count]
```

The `ping` command determines whether the Sun Fire B10n blade has connectivity or whether a host is available on the network. The command output shows whether the response was received, that is, the host exists on the network.

If the host is not responding then `ping` displays this message:

```
no answer from hostname
```

If the host is available on the network then `ping` displays this message:

```
hostname is alive
```

TABLE 4-4 describes the parameters for sending a `ping` request.

TABLE 4-4 Parameters for Sending a `ping` Request

Variables	Description
<code>ip-addr</code>	IP address of a host on the network
<code>hostname</code>	Name of a host on the network
<code>packet-count</code>	Number of tries

▼ To Unconfigure a Network Interface

- As `admin` in `config` mode, use the following command:

```
puma(config){admin}# no ip interface {0|1}
```

This command unconfigures a network interface on the Sun Fire B10n blade.

▼ To Configure a DNS Server

You can configure both a primary and a secondary Domain Name Server (DNS) for the Sun Fire B10n blade. When supplied with a hostname, the DNS server resolves it and obtains the corresponding IP address.

- As `admin` in `config` mode, use the following command:

```
puma(config){admin}# dns server ip-addr {primary|secondary}
```

▼ To Remove a DNS Server

- As `admin` in `config` mode, use the following command:

```
puma(config){admin}# remove dns server ip-addr
```

▼ To Configure the DNS Suffix

- As admin in config mode, use the following command:

```
puma(config){admin}# dns suffix suffix-name
```

This sets the suffix to be added to the hostnames before resolution with a DNS resolver to get the IP address.

Note – For example, a DNS suffix might be mycompany.com.

▼ To Show DNS Use

- As any user, use the following command:

```
puma{user}# show network
```

▼ To Unconfigure the DNS Suffix

- As admin in config mode, use the following command:

```
puma(config){admin}# no dns suffix
```

▼ To Configure the Default Gateway

- As admin in config mode, use the following command:

```
puma(config){admin}# default gateway ip-addr
```

Where *ip-addr* is the gateway IP address.

▼ To Unconfigure the Default Gateway

- As admin in config mode, use the following command:

```
puma(config){admin}# no default gateway
```

▼ To Set the Default Hostname

- As admin in config mode, use the following command:

```
puma(config){admin}# default hostname hostname
```

This example configures the default hostname as B10n-no-1:

```
puma(config){admin}# default hostname B10n-no-1
```

After you set up the hostname, your CLI prompt displays the hostname:

```
B10n-no-1(config){admin}#
```

▼ To Show the Network Configurations

- As any user, use the following command:

```
puma{user} # show network
```

This command returns the following information:

- IP address and netmask on the two interfaces
- The default gateway
- The DNS server and suffix configurations
- The management VLAN
 - If set, what is the value
 - Whether it is enabled or not
- The default data VLAN
 - If set, what is the value

- Whether it is enabled or not.
- The server monitoring information

▼ To Show ARP Entries

1. As admin, use the following command:

```
puma{admin}# show arp
```

The output from this command shows all the entries in the ARP table.

The following example shows a typical output from the `show arp` command:

```
LINK LEVEL ARP TABLE
destination      gateway          flags Refcnt  Use      Interface
-----
192.50.50.11     00:03:af:26:73:07405  0      35330     iq0
192.50.50.12     00:03:af:26:97:fb405  1      16653     iq0
-----
```

Note – In the ARP table the gateway and flags columns are improperly shown. In the example above, 405 should align under the flags heading. However, the gateway and flags fields are merged. The incorrect merging is a limitation of the underlying software.

See “To Configure the Subnet Mask for a VIP” on page 73 to create a VIP subnet mask.

▼ To List the VIPs Configured

- As any user, use the following command:

```
puma{user}# show vip
```

This command lists all the VIPs configured on the Sun Fire B10n blade.

▼ To Show All End Points

- As any user, use the following command:

```
puma{user}# show end-points
```

This command lists all the end points in the system. It lists all the 3 tuples, their type, for example, tracking end points, service end points, and so on, and the name of the service to which they belong.

Monitoring

The blade servers are monitored for health and connectivity. This involves collecting data such as server response time (or server load), network latency to a server, whether the server is up, whether the network (either the network connection between the content load balancing blade and the server or the network interface on the server) is up, and so on. The actual data collection on each server is performed by a control module residing on the server. The statistics are obtained by the management module on the Sun Fire B10n blade using SNMP.

▼ To Set Up Server Monitoring

- As admin in config mode, use the following command:

```
puma(config){admin}# server-monitor interval interval-val [max-try max-try-val]
```

The config `server-monitor` command configures the monitoring parameters for detecting loss of connectivity to a back end server or a failure of the server itself.

TABLE 4-5 describes the parameters for setting up a server for monitoring.

TABLE 4-5 Parameters and Variables for Setting Up a Server for Monitoring

Parameters and Variables	Description
<code>interval</code>	(Optional) Qualifier for the value specified in the <i>monitoring-interval</i> argument.
<code>interval-val</code>	(Optional) The time interval (in seconds) in which the monitoring messages are sent. The default value is 3.
<code>max-try</code>	(Optional) Qualifier for the value specified in the <i>max-try-val</i> argument.
<code>max-try-val</code>	(Optional) The maximum number of tries for the monitoring messages before marking a server down. The default value is 5.

▼ To Set Up Application Monitoring

- As admin in config mode, use the following command:

```
puma(config){admin}# service app-monitor name service-name interval interval-val max-try max-try-val {{proto {tcp | http}} | {script filename}}
```

The `config service app-monitor` command configures the monitoring parameters for detecting the health of the application for a particular service.

TABLE 4-6 describes the parameters for this command:

TABLE 4-6 Parameters for Setting Up Application Monitoring

Variable	Description
<code>name</code>	Qualifier for the value specified in the <i>service-name</i> argument.
<code>service-name</code>	Name of the service for which application monitoring is configured.
<code>interval</code>	Qualifier for the value specified in the <i>interval-val</i> argument.
<code>interval-val</code>	Interval in seconds which the monitoring is to be performed. Valid values are between 15 to 3600. The default value is 30.
<code>max-try</code>	Qualifier for the value specified in the <i>max-try-val</i> argument.

TABLE 4-6 Parameters for Setting Up Application Monitoring (*Continued*)

Variable	Description
<i>max-try-val</i>	Number of consecutive times a valid server response is not received before marking the server down. The default value is 3.
proto	Qualifier for the protocol value.
http tcp	Protocol to use for monitoring. For HTTP, the response code of 200 is checked. If a file is not found or if any other internal server error happens the server is marked as down. For TCP, a connection is made and immediately closed and no data is transferred. This will detect if the specified application is running or not.
script	Qualifier for the value specified in the <i>filename</i> argument.
<i>filename</i>	Script file that can be specified, instead of a protocol, for doing the monitoring.

▼ To Configure Application Monitoring Parameters

- As admin in config mode, use the following command:

```
puma(config){admin}# service app-monitor-param name service-name param param-name param-value
```

This command configures parameters for the specified protocol. HTTP supports setting of the URL.

TABLE 4-8 describes the parameters for this command:

TABLE 4-7 Parameters for Configuring Application Monitoring

Variable	Description
name	Qualifier for the value specified in the <i>service-name</i> argument.
<i>service-name</i>	Name of the service for which application monitoring parameters is to be configured.

TABLE 4-7 Parameters for Configuring Application Monitoring (*Continued*)

Variable	Description
<i>param</i>	Qualifier for the value specified in the <i>param-name param-value</i> argument.
<i>param-name</i>	Name of the parameter. For HTTP, the only parameter supported is a URL. This parameter enables specifying the URL to use in the HTTP GET request.
<i>param-value</i>	The corresponding parameter value. For HTTP URL parameter, this is the actual URL to use in the HTTP GET request.

▼ Enable or Disable Application Monitoring for a Specified Service

- As admin in config mode, use the following command:

```
puma(config){admin}# [no] enable service app-monitor name service-name
```

Where *service-name* is the name of the service for which application monitoring is to be enabled or disabled.

Configuring SSL Device Entries

The Sun Fire B10n blade can be configured to work in conjunction with one or more SSL proxy blades for content load balancing SSL traffic. You must configure an SSL entry on the content load balancing blade for each SSL proxy blade in the system.

▼ To Add an SSL Device

The `config ssl name` command adds an entry for an SSL proxy blade on the content load balancing blade with at least one interface configured.

- As admin in config mode, use the following command:

```
puma(config){admin}# ssl name ssl-device-name {ssl-ip-1 | hostname-1} [{ssl-ip-2 | hostname-2}]
```

Examples

The first example creates an SSL device `ssl1`, with an IP address of 192.50.50.12.

```
puma{config}{admin}# ssl name ssl1 192.50.50.12
```

The second example creates an SSL device `ssl2`, with an IP address of 192.50.50.14 and 192.50.50.15.

```
puma{config}{admin}# ssl name ssl1 192.50.50.14 192.50.50.15
```

TABLE 4-8 describes the parameters for adding SSL device configurations.

TABLE 4-8 Parameters for Adding SSL Device Configurations

Variable	Description
<i>ssl-device-name</i>	Name of the SSL device entry.
<i>ssl-ip-1</i>	IP address of the SSL device on one interface.
<i>hostname-1</i>	Host name of the SSL device on the same interface.
<i>ssl-ip-2</i>	(Optional) IP address of the SSL device on the other interface.
<i>hostname-2</i>	(Optional) Host name of the SSL device on the same interface.

▼ To Remove an SSL Device

The `remove ssl name` command removes an SSL device entry.

- As admin in config mode, use the following command:

```
puma(config){admin}# remove ssl name {ssl-device-name}
```

Example

The following example removes an SSL device `ssl3`.

```
puma{config}{admin}# remove ssl name ssl3
```

Note – If the SSL device has either of its interfaces included in any service, then this command fails.

▼ To Add a Port Pair to an SSL Device Entry

- As admin in config mode, use the following command:

```
puma(config){admin}# ssl port-pair ssl-device-name secureport secure-port-num clearport  
clear-port-num
```

Example

The following example adds a port pair to an SSL device `ssl1`, with secure port as 443 and clear port as 880.

```
puma{config}{admin}# ssl port-pair ssl1 secureport 443 clearport 880
```

The `ssl port-pair` command configures an SSL device entry with a secure port and the corresponding clear port, that is, a port pair configuration.

Note – A maximum of four such port pairs can be added to an SSL device entry. Each of the eight ports must be unique. Maximum value of each port is 1023.

TABLE 4-9 describes the parameters for adding and removing port pairs.

TABLE 4-9 Parameters for Adding and Removing Port Pairs

Parameter	Description
<i>ssl-device-name</i>	Name of the SSL device entry.
secureport	Qualifier for the secure port.
<i>secure-port-num</i>	The secure port number. This is the port at which a secure service configured with this SSL device accepts SSL encrypted traffic from the client.
clearport	Qualifier for the clear port.
<i>clear-port-num</i>	The clear port number. This is the port to which this SSL device sends the traffic after decryption.

▼ To Remove a Port Pair from an SSL Device Entry

- As admin in config mode, use the following command:

```
puma(config){admin}# remove ssl port-pair {ssl-device-name} secureport {secure-port-num}  
clearport {clear-port-num}
```

Example

The following example removes a port pair from an SSL device `ssl1`.

```
puma{config}{admin}# remove ssl port-pair ssl1 secureport 443 clearport 880
```

The `remove ssl port-pair` command removes a port pair configuration from an SSL device entry.

Note – If the SSL device has either of its interfaces included in any service, then this command fails.

▼ To Add an Interface to an SSL Device Entry

- As admin in config mode, use the following command:

```
puma(config){admin}# ssl if ssl-device-name {ssl-ip|hostname}
```

Example

The following example adds interface 192.50.50.13 to an existing SSL device `ssl1`.

```
puma{config}{admin}# ssl if ssl1 192.50.50.13
```

The `ssl if` command configures an interface for an SSL device entry.

Note – A maximum of two interfaces can be configured for an SSL device entry at any time.

TABLE 4-10 describes the variables for adding or removing an interface.

TABLE 4-10 Parameters for Adding or Removing an SSL Device Interface

Parameter	Description
<i>ssl-device-name</i>	Name of the SSL device entry.
<i>ssl-ip</i>	The IP address of the SSL device interface.
<i>hostname</i>	The host name of the SSL device interface.

▼ To Remove an Interface from an SSL Device Entry

- As admin in config mode, use the following command:

```
puma(config){admin}# remove ssl if ssl-device-name {ssl-ip|hostname}
```

Example

The following example removes an interface 192.50.50.13 from an SSL device `ssl1`.

```
puma(config){admin}# remove ssl if ssl1 192.50.50.13
```

The `remove ssl if` command removes an interface from an SSL device entry.

Note – An SSL device entry must have at least one interface configured. It is not possible to remove an interface that is included in one or more services.

▼ To Enable an SSL Device Entry

The `enable ssl name` command enables an SSL device entry.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# enable ssl name {ssl-device-name}
```

By default, an SSL entry is enabled when it is created.

▼ To Disable an SSL Device Entry

The `no enable ssl name` command disables an SSL device entry.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# no enable ssl name {ssl-device-name}
```

▼ To Show the Configured SSL Devices

- As any user, use the following command:

```
B10n {user} # show ssl [ssl-device-name]
```

You can use this command to display the SSL devices configured for the content load balancing blade. By specifying the *ssl-device-name*, you can show one specific blade.

Configuring Multiple SSL Devices

To configure multiple SSL devices, repeat the steps under “Configuring SSL Device Entries” on page 61 for each SSL device to be added. All the configured SSL devices can be displayed using the `show ssl` command.

Configuring the Content Load Balancing Blade

Note – Layer 7 load balancing is for HTTP (web-based) services only.

For Layer 7 load balancing, some TCP parameters must be configured on the content load balancing blade. The TCP stack in each of the blade servers being load balanced must be configured with the same parameters.

Note – For each of these parameters, the content load balancing blade starts up with a set of defaults that match those on the servers at the time of deployment.

▼ To Set the TCP Parameters

The `config default tcp-params` command sets the TCP parameters on the content load balancing blade to be used for TCP connections that are Layer 7 load balanced. These parameters must be set for each blade and serve as defaults. They can be overwritten for each individual service if required.)

- As admin in config mode, use the following command:

```
puma(config){admin}# default tcp-params [window tcp-window] [window-scale tcp-ws-factor] [ts] [sack]
```

The following example sets the TCP default window to 2048, the window scaling factor to 1, and the TCP timestamp and SACK options:

```
puma(config){admin}# default tcp-params window 2048 window-scale 1 ts sack
```

TABLE 4-11 describes the options for setting the TCP parameters for the content load balancing blade.

TABLE 4-11 Options for Setting the TCP Parameters

Parameter	Description
<code>window</code>	(Optional) Qualifier for the value specified in the <code>tcp-window</code> argument.
<code>tcp-window</code>	(Optional) The TCP window size to use in the SYN.
<code>window-scale</code>	(Optional) Qualifier for the value specified in the <code>tcp-ws-factor</code> argument.
<code>tcp-ws-factor</code>	(Optional) The TCP window scaling factor to advertise in the SYN.
<code>ts</code>	(Optional) TCP timestamp is enabled. If the parameter is omitted then TCP timestamp is disabled.
<code>sack</code>	(Optional) TCP SACK is enabled. If the parameter is omitted then TCP timestamp is disabled.

Defaults

`tcp-window` defaults to 8192. `tcp-ws-factor` defaults to 0, that is, the window scaling option is not supported. The SACK option is supported by default, but the timestamp option is not.

Examples

The first example sets the TCP window to 2048:

```
puma(config){admin}# default tcp-params window 2048
```

The following example scales the window to 1 and includes the timestamp option:

```
puma(config){admin}# default tcp-params window-scale 1 ts
```

The third example adds only the timestamp and SACK options:

```
puma(config){admin}# default tcp-params ts sack
```

▼ To Set Parameters for TCP Connection Handoff

For TCP load balancing services, the Sun Fire B10n blade performs a connection handoff to the back end servers. The maximum number of times the handoff message is to be retransmitted to a server before trying a new server is a parameter that is set to a default value on the content load balancing blade. The default can be changed as needed. This feature applies to Layer 7 load balancing services only.

The `default tcp-handoff-params` command sets the default TCP connection handoff parameters on the content load balancing blade. These parameters are set for each blade and serve as defaults. They can be overwritten for a service if required.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# default tcp-handoff-params {max-open-resend}
```

Where *max-open-resends* is the maximum number of times the OPEN message is retransmitted to the same server before load balancing again. The default value for *max-open-resends* is 5.

Example

The following example sets the value for the maximum retransmissions of the handoff message.

```
puma(config){admin}# default tcp-handoff-params 7
```

▼ To Show All the Default TCP Parameters Settings

- **As any user, use the following command:**

```
B10n {user} # show default tcp
```

The `show default tcp` command shows default TCP parameters, TCP DoS defense parameters, and TCP connection handoff parameters configured on the content load balancing blade.

Load Balancing Service Configuration and Management

A load balancing service on Sun Fire B10n blade is characterized by a VIP, a port, and a protocol, the interface on content load balancing blade to which the service is bound, SSL support, the load balancing layer and, if applicable, the load balancing protocol. Other configurations can be added incrementally to a service.

Creating a Load Balancing Service

The `service` command creates an entry for a load balancing service on the blade. Once you have created the service, you can add more configurations to it as needed.

Note that before a created service can be functional, a minimum of two commands must be executed for the service. First, a default group of servers and a load balancing scheme must be specified by using the `config service lb-group default` command. Second, the service (which is created in a disabled state) must be enabled by the `enable service` command.

The available load balancing schemes are as follows:

- Round-robin (`round-robin`)
- Weighted round-robin (`wt-round-robin`)
- Static (`static`)
- Weighted least connections (`wt-least-conn`)
- Response time (`response-time`)

For rule-based load balancing, the service must be linked to one or more blade servers, a load balancing scheme, and optionally, a load balancing rule by the `config service lb-group` command.

Note – Currently, the only Layer 7 protocol that can be Layer 7 load balanced is HTTP.

Note – In the absence of an SSL proxy, SSL traffic can be load balanced only on Layer 4.

▼ To Create a Load Balancing Service

- As admin in config mode, use the following command:

```
puma(config){admin}# service name service-name vip {VIP-address | hostname}:port-  
num:{tcp|udp} [ssl decrypted-port] interface {0|1} [lb-layer {4|7}] [L7-proto {http|ftp}]
```

Note – When adding an SSL service, using the same VIP address with a different port, but the same SSL port is not allowed. The new SSL service must have a unique port number. For example, if an initial SSL service is running on SSL port 880, you must specify a different SSL port number for each new SSL service such as SSL port 881, 882, and so on.

TABLE 4-12 describes the parameters for creating a load balancing service.

TABLE 4-12 Parameters for Creating a Load Balancing Service

Parameter	Description
<code>service-name</code>	Qualifier for the service name.
<i>service-name</i>	Configured name of the service (ASCII string).
<code>vip</code>	Qualifier for the virtual service address.
<i>VIP-address</i>	Destination IP address for the service.

TABLE 4-12 Parameters for Creating a Load Balancing Service (Continued)

Parameter	Description
<i>hostname</i>	Destination host name for the service.
<i>port-num</i>	Destination TCP/UDP port number for the service.
<i>tcp</i>	The Layer 4 protocol is TCP.
<i>udp</i>	The Layer 4 protocol is UDP.
<i>ssl</i>	(Optional) Specifies this service end point as an SSL end point.
<i>decrypted-port</i>	(Optional) Specifies the decrypted port number for traffic coming back from the SSL device. The maximum allowed value is 1023.
<i>interface</i>	Uses the value specified in the following argument.
0	This service point is bound to network interface number 0 on the content load balancing blade.
1	This service point is bound to network interface number 1 on the content load balancing blade.
<i>lb-layer</i>	(Optional) Qualifier for the OSI layer at which load balancing is to be performed.
4	Load balancing is performed based on the Layer 4 fields of the incoming packets.
7	Load balancing is performed based on the Layer 7 fields of the incoming packets.
<i>L7-PROTO</i>	(Optional) Qualifier for the protocol that is Layer 7 load balanced.
<i>http</i>	The Layer 7 protocol to be load balanced is HTTP.
<i>ftp</i>	The Layer 7 protocol to be load balanced is FTP.

Defaults

The default load balancing is at Layer 4. The default Layer 7 protocol is HTTP. Each service must have a unique 3-tuple (*vip*, *port-num*, {*tcp|udp*}). If your service is an SSL service, it should also have a unique decrypted 3-tuple (*vip*, *decrypted-port*, {*tcp|udp*}).

Examples

The first example creates a service named *SVC0*, with a VIP of 192.50.50.1, on port 80, using the TCP scheme. *SVC0* is a Layer 4 load balanced TCP service.

```
puma(config){admin}# service name SVC0 vip 192.50.50.1:80:tcp
interface 0
```

The second example, creates a service named `svc2`, which is a Layer 4 load balanced SSL service bound to interface 0 of the content load balancing blade. The SSL decrypted port is 880.

```
puma(config){admin}# service name svc2 vip 192.50.50.1:443:tcp ssl
880 interface 0
```

The last example creates a service named `SVC1`, which is a Layer 7 load balanced HTTP service.

```
puma(config){admin}# service name SVC1 vip 192.50.50.1:8080:tcp
interface 1 lb-layer 7 L7-proto http
```

▼ To Configure the Subnet Mask for a VIP

- As admin in config mode, use the following command:

```
puma(config){admin} # vip-netmask {ip-addr | hostname} mask netmask
```

Note – This command is used to configure the subnet masks for VIPs which have already been created in the system using the `config service name`, `config service point` or `config service tracking` commands.

TABLE 4-13 describes the parameters for configuring the subnet mask for a VIP.

TABLE 4-13 Parameters for Configuring the Subnet Mask for a VIP

Parameter	Description
<i>ip-addr</i>	VIP address.
<i>hostname</i>	Host name for the VIP.
<i>mask</i>	Qualifier for the net mask.
<i>netmask</i>	Subnet mask. This should be in the xxx.xxx.xxx.xxx format.

▼ To Add SSL Devices to a Service

The `service ssl` command adds one or more SSL devices to the SSL load balancing group of a service.

- As admin in config mode, use the following command:

```
puma(config){admin}# service ssl service-name ssl ssl-device-name: {active|standby} [ssl-device-name: {active|standby}]...
```

Examples

The following example adds an SSL device named `ssl1` to the Service `SVC1`.

```
puma{config}{admin}# service ssl SVC1 ssl ssl1:active
```

To add a second SSL device to the same service, the command should be invoked again for the second SSL device.

```
puma{config}{admin}# service ssl SVC1 ssl ssl2:active
```

Both devices can be added in one command also.

```
puma{config}{admin}# service ssl SVC1 ssl ssl1:active ssl2:active
```

One device can be added as Active and another device can be added as Standby.

```
puma{config}{admin}# service ssl SVC1 ssl ssl1:active ssl2:standby
```

TABLE 4-14 describes the parameters for adding one or more SSL devices to the SSL load balancing group of a service.

TABLE 4-14 Parameters for Adding SSL Devices to a Service

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<i>ssl</i>	Qualifier for the SSL device argument.
<i>ssl-device-name</i>	Name of the SSL proxy blade.
<i>active</i>	The SSL proxy blade is added in the active mode.
<i>standby</i>	The SSL proxy blade is added in the standby mode.

Usage Guidelines

The `service ssl` command should be invoked at least once for any service that has one or more end points enabled for SSL.

If additional SSL devices are created after executing the `service ssl` command, execute this command again with the new SSL device name.

```
puma(config){admin}# service ssl service-name ssl ssl-device-name:{active|standby} [ssl-device-name:{active|standby}]
```

▼ To Remove SSL Devices From a Service

- As admin in config mode, use the following command:

```
puma(config){admin}# remove service ssl service-name ssl ssl-device-name [ssl-device-name]
```

Example

The following example removes an SSL device `ssl1` from the service `SVC1`.

```
puma{config}{admin}# remove service ssl SVC1 ssl ssl1
```

This command removes one or more SSL devices from the SSL load balancing group of a service.

TABLE 4-15 describes the parameters for removing one or more SSL devices from the SSL load balancing group of a service.

TABLE 4-15 Parameters for Removing SSL Devices from a Service

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<code>ssl</code>	Qualifier for the SSL device argument.
<i>ssl-device-name</i>	Name of the SSL proxy blade.

Note – If a service is SSL enabled, it should have at least 1 active SSL device. So it is not possible to remove the last SSL device from an SSL enabled service.

▼ To Set SSL Devices in a Service as Active or Standby

- As admin in config mode, use the following command:

```
puma(config){admin}# modify service ssl mode service-name ssl ssl-device-name [ssl-device-name...] mode {active|standby}
```

This command sets one or more SSL devices in the SSL load balancing group of a service as either active or standby.

TABLE 4-14 describes the parameters for modifying one or more SSL devices in the SSL load balancing group of a service as either active or standby.

TABLE 4-16 Parameters for Modifying SSL Devices in a Service

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<i>ssl</i>	Qualifier for the SSL device argument.
<i>ssl-device-name</i>	Name of the SSL proxy blade.
<i>mode</i>	Qualifier for the device mode.
<i>active</i>	Set the device as active for the service.
<i>standby</i>	Set the device as standby for the service.

Note – If a service is SSL enabled, it should have at least 1 active SSL device. So it is not possible to configure the last SSL device as stand by in an SSL enabled service.

▼ To Add a Default Load Balancing Group to a Load Balancing Service

- As admin in config mode, use the following command:

```
puma(config){admin}# service lb-group default service-name server {ip-addr | hostname};port:protocol:weight:active [{ip-addr | hostname};port:protocol:weight:active...] [scheme {round-robin | wt-round-robin | static | wt-least-conn | response-time}]
```


The `config service lb-group default` command configures one or more servers to which a request for a service is directed if it does not match any of the rules in any of the load balancing groups configured for that service. The load balancing scheme is also configured. This is the default load balancing group for the service. All the load balancing group commands can be applied to this group with the load balancing group name specified as the default.

TABLE 4-17 describes the parameters for this command.

TABLE 4-17 Parameters for Adding a Default Load Balancing Group to a Service

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<i>server</i>	Qualifier for the server or real service argument.
<i>ip-addr</i>	IP address of the back end server.
<i>hostname</i>	Host name of the back end server.
<i>port</i>	Port on the back end server where this service can be provided. If specified as 0, it means that this is just a server and not a real service protocol.
<i>protocol</i>	Corresponding protocol on the back end server.
<i>weight</i>	Weight for this blade server. Valid only if the load balancing scheme used is weighted round robin. Otherwise, this is ignored. 65535 is the maximum weight supported.
<i>active</i>	Specifies the blade server as active for this load balancing group if the value is 1, standby if the value is 0.
<i>scheme</i>	(Optional) Qualifier for the load balancing scheme.
<i>round-robin</i>	(Optional) Load balancing scheme is round robin.
<i>wt-roundrobin</i>	(Optional) Load balancing scheme is weighted round robin.
<i>wt-least-conn</i>	(Optional) Load balancing scheme is weighted least connections.
<i>response-time</i>	(Optional) Load balancing scheme is response time.
<i>static</i>	(Optional) Load balancing scheme is static load balanced, where the server is chosen by a hash function. Used when the service is configured for UDP.

Note – Port NAT is not supported at this time. So the *port* and *protocol* field values are ignored.

Defaults

The default load balancing scheme is round robin for a TCP service. When the load balancing scheme is weighted round robin and the weight is specified as 0 for a server, then the default weight is 1. For round robin and static load balancing schemes the weight field is ignored.

Usage Guidelines

Invoke this command at least once for any service after the service is created and before it starts accepting connections.

For a UDP service, the only load balancing scheme supported is static.

Examples

The first example uses the default scheme and only one server.

```
puma(config){admin}# service lb-group default SVC1 server  
192.50.50.201:0:tcp:5:1
```

The following example sets three servers, uses the TCP protocol, and specifies the scheme as weighted round robin.

```
puma(config){admin}# service lb-group default SVC1 server  
192.50.50.201:0:tcp:5:0 192.50.50.202:0:tcp:10:1 192.50.50.203:0:tcp:7:1  
scheme wt-round-robin
```

▼ To Set the TCP Parameters for a Service

- As admin in config mode, use the following command:

```
puma(config){admin}# service tcp-params service-name [window tcp-window] [window-  
scale tcp-ws-factor] [ts] [sack]
```

TABLE 4-18 describes the TCP parameters to set for a service.

TABLE 4-18 TCP, Parameters for a Service

Parameter	Description
<i>service-name</i>	The name of the service to set the TCP parameters.
<i>window</i>	Qualifier for the value specified in the <code>tcp-window</code> argument.
<i>tcp-window</i>	The TCP window size to use in the SYN.
<i>window-scale</i>	(Optional) Qualifier for the value specified in the <code>tcp-ws-factor</code> argument.
<i>tcp-ws-factor</i>	(Optional) The TCP window scaling factor to advertise in the SYN.
<i>ts</i>	(Optional) TCP timestamp is enabled. If the parameter is omitted then TCP timestamp is disabled.
<i>sack</i>	(Optional) TCP SACK is enabled. If the parameter is omitted then TCP timestamp is disabled.

The `service tcp-params` command overwrites the default TCP settings on the load balancer and is valid only if the protocol for the service is TCP and the load balancing is done at Layer 7.

For most cases, use the TCP parameters set by the `config default tcp-params` command for each content load balancing blade since those are the parameters with which all the back end servers served by the content load balancing blade are configured. If this command is invoked for any service to change these defaults, the network administrator must modify the TCP parameters of the servers accordingly. But if any one of these servers is included in another service with different TCP parameters, then this command fails.

Defaults are the values configured for the content load balancing blade.

Usage Guidelines

For most cases, the TCP parameters set by the `default tcp-params` command on each content load balancing blade should be used. Those are the parameters with which all the back end servers served by the content load balancing blade are configured. If this command is invoked at all for any service to change these defaults, the network administrator should ensure that the servers added to this service have their TCP parameters modified accordingly. But if any one of these servers is included in another service with different TCP parameters, then this command fails.

Example

```
puma(config){admin}# service tcp-params SVC1 window 2048 window-scale 1 ts
```

▼ To Set Parameters for TCP Connection Handoff for a Service

The `service tcp-handoff-params` command modifies the default TCP connection handoff parameters for a service. This command is valid only if the protocol for the service is TCP.

- As admin in config mode, use the following command:

```
puma(config){admin}# service tcp-handoff-params service-name max-open max-open-resends
```

Where:

service-name is the name of the service

max-open uses the value specified in the *max-open-resends* argument.

max-open-resends is the maximum number of times the OPEN message is retransmitted to the same server before load balancing again.

Defaults

The default is the same as that set for the content load balancing blade.

Example

The following example sets the maximum number of times the OPEN message is retransmitted to the same server before load balancing again to four times for the service SVC1.

```
puma(config){admin}# service tcp-handoff-params SVC1 max-open 4
```

▼ To Add a Service Point to a Service

The `service point` command adds one or more IP address, port, and protocol combinations to a given service, making the service multihomed.

If the VIP is already bound to an interface in any service, be sure to specify that interface. Two service end points on a given service cannot have the same VIP. The protocol for every added end point should be the same as the service protocol.

- As admin in config mode, use the following command:

```
puma(config){admin}# service point service-name point {ip-addr | hostname};port-  
num:proto:ssl:decrypted-port:ber:interface [{ip-addr | hostname};port-num:proto:ssl:  
ssl-port-number:interface...]
```

TABLE 4-19 describes the service point parameters available for a service.

TABLE 4-19 Service Point Parameters

Parameter	Description
<i>service-name</i>	Name of the service.
<i>point</i>	Qualifier for the service point.
<i>ip-addr</i>	IP address of the service point.
<i>hostname</i>	Host name of the service point.
<i>port-num</i>	Corresponding port number of the service point. The valid range of this parameter is 0-65535.
<i>proto</i>	Corresponding protocol of the service point: TCP or UDP.
<i>ssl</i>	If 1, specifies the added service end point as SSL enabled, if 0, no SSL support provided for service point.
<i>decrypted-port</i>	The decrypted port number to which the traffic from the SSL device is destined. Maximum allowed value is 1023.
<i>interface</i>	Specifies the interface on the content load balancing blade on which this service point is bound. Can have the following values: 0: The service point bound to network interface number 0 on a content load balancing blade. 1: The service point bound to network interface number 1 on a content load balancing blade.

Note – A maximum of three end points are allowed for each service. This includes the end point with which the service was created.

Note – If the service was originally non-SSL and SSL is enabled with the end point added using the `service point` command, an SSL device should be added to the service using the `conf service ssl` command.

Examples

The first example adds two service points to the service `svc1`, both using TCP protocol, on port 80, with no SSL support.

```
puma(config){admin}# service point svc1 point 192.50.51.1:80:tcp:0:0:0  
192.50.51.2:80:tcp:0:0:0
```

The following example adds one service point to the service `svc2` on interface 0, port 80, protocol TCP, with SSL support and the decrypted port specified as 880.

```
puma(config){admin}# service point svc2 point 192.50.51.3:80:tcp:1:880:0
```

▼ To Remove a Service Point From a Service

The `remove service point` command removes one or more service points from a given service.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# remove service point service-name point {ip-addr | hostname}:port-  
num:proto [{ip-addr | hostname}:port-num:proto...]
```

TABLE 4-20 describes the service point parameters to be removed from a service.

TABLE 4-20 Service Point Parameters

Parameter	Description
<i>service-name</i>	Name of the service.
<i>point</i>	Qualifier for the service point.
<i>ip-addr</i>	IP address of the service point.

TABLE 4-20 Service Point Parameters

Parameter	Description
<i>hostname</i>	Host name of the service point.
<i>port-num</i>	Corresponding port number of the service point.
<i>proto</i>	Corresponding protocol of the service point.

Example

This example removes a service point with VIP 192.50.51.1, port 80, and protocol TCP from the service SVC1.

```
puma(config){admin}# remove service point SVC1 point 192.50.51.1:80:tcp
```

▼ To Configure a Service for Client IP or Subnet-Based Persistence

The `service ip-persist` command configures a service for persistence based on the client IP address or subnet.

If configured for client IP persistence, all traffic to this service coming from the same client IP (or same subnet in case a mask is specified) is sent to the same back end server. The timer specifies the inactivity interval after which this persistence ceases to exist, that is, subsequent traffic from the same client IP (or subnet) to this service is load balanced to another blade server.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# service ip-persist service-name [mask mask-len] [timeout timeout-val]
```

TABLE 4-21 describes the parameters for configuring a service for persistence.

TABLE 4-21 Parameters for Configuring a Service for Persistence

Parameter	Description
<code>service-name</code>	The name of the service entry to be configured for client IP persistence.
<code>mask</code>	(Optional) The client IP mask used for persistence.
<code>mask-len</code>	(Optional) The number of bits to be masked from the right. The valid range is 0 to 31.
<code>timeout</code>	(Optional) Uses the value specified in the <code>timeout-val</code> argument. Short timeout values do not have the expected result. Timeout values must be at least 15–20 minutes, and the granularity of timing out entries is in the order of 10 minutes.
<code>timeout-val</code>	(Optional) The inactivity time (in minutes) after which a persistence is removed. The valid range is 0 to 1092.

Defaults

The default behavior is IP persistence for a specific client IP (when a subnet is not specified, that is, `mask-len = 0`). The default timeout value is the same as that configured for service point tracking if such a configuration exists, otherwise the timeout is five minutes.

Note – Short timeout values do not have the expected result. Timeout values should be at least 15–20 minutes, and the granularity of timing out entries is in the order of 10 minutes.

Examples

The first example sets service IP persistence for the service named SVC1.

```
puma(config){admin}# service ip-persist SVC1
```

The following example sets service IP persistence for the service SVC1 using mask 16 and a timeout of 20 minutes.

```
puma(config){admin}# service ip-persist SVC1 mask 16 timeout 20
```


▼ To Remove Client IP or Subnet Based Persistence from a Service

When client IP persistence is removed from a service, then any new connections to the service are load balanced again.

- As admin in config mode, use the following command:

```
puma(config){admin}# no service ip-persist service-name
```

Example

```
puma(config){admin}# no service ip-persist SVC1
```

Configuring a Service for Service Point Tracking

The `service tracking` command configures a service for tracking one or more service points with or without the destination VIPs specified.

If configured for service point tracking, all traffic to this service coming from the same client IP and destined to any of the tracking service points specified in the configuration is sent to the same back end server. The timer specifies the inactivity interval after which this persistence ceases to exist, that is, subsequent traffic from the same client IP, destined to any of the tracking service points configured is load balanced to another back end server. Service point tracking is a special case of client IP-based persistence.

Note – The timeout value may overwrite the one specified by the `config service ip-persist` command if that has been already invoked for this service as only one persistence timer is maintained for each service.

In case the VIP to track is not specified (that is, specified as 0), the service performs port tracking on the specified service points.

Note – End point tracking is added only to the primary VIP, that is, the VIP end point with which the service was created. Port tracking is added to all the VIP end points of a multihomed service.

Note – The maximum number of service points that can be added for tracking is five.

▼ To Configure a Service for Service Point Tracking

- As admin in config mode, use the following command:

```
puma(config){admin}# service tracking service-name track [ip-addr | hostname]:port:proto
[ip-addr | hostname]:port:proto] timeout timeout-val
```

TABLE 4-22 describes the parameters for configuring a service for service point tracking.

TABLE 4-22 Parameters for Configuring a Service for Tracking

Parameter	Description
<i>service-name</i>	Name of the service entry to be configured for service point tracking.
<i>track</i>	Qualifies the service point that should track the primary service point.
<i>VIP-address</i>	VIP address of the service point to track the primary. If 0, then only the port (and protocol) is tracked.
<i>hostname</i>	Host name of the service point to track the primary. If 0, then only the port (and protocol) is tracked.
<i>port</i>	Port number of the service point to track the primary.
<i>proto</i>	Protocol of the service point to track the primary.
<i>timeout</i>	(Optional) Uses the value specified in the <i>timeout-val</i> argument.
<i>timeout-val</i>	(Optional) The inactivity time (in minutes) after which tracking stops.

Defaults

The default behavior is port tracking on the specified service points (when the VIP is specified as 0). The default timeout value is the same as that configured for client IP persistence if such a configuration exists, else five minutes.

Examples

The first example sets port tracking for the service named `SVC1`, at port 443, using the TCP protocol.

```
puma(config){admin}# service tracking SVC1 track 0:443:tcp
```

The following example sets end point tracking entries for the service named `SVC1`: One tracking end point is given by VIP 188.88.8.1, port 9090, protocol TCP. The other end point has VIP 177.77.7.1. port 80, using the TCP protocol.

```
puma(config){admin}# service tracking SVC1 track 188.88.8.1:9090:tcp  
177.77.7.1:80:tcp timeout 20
```

▼ To Remove Tracking Service Point from a Service

- As admin in config mode, use the following command:

```
puma(config){admin}# remove service tracking service-name track {ip-addr |  
hostname}:port:proto
```

This command removes a tracking service point from a service. See TABLE 4-22 for descriptions of the parameters.

Examples

The first example removes port tracking for the service named `SVC1`, at port 443, using the TCP protocol.

```
puma(config){admin}# remove service tracking SVC1 track 0:443:tcp
```

The following example removes two end point tracking end points from the service `SVC1`.

```
puma(config){admin}# remove service tracking SVC1 track 188.88.8.1:9090:tcp  
177.77.7.1:80:tcp
```

▼ To Configure a Service for Cookie-Based Persistence

The purpose of the `service cookie-persist` command is to handle cookies embedded in a packet and ensure persistence across connections for an application offered by a particular blade server in this service.

1. As admin in config mode, use the following command:

```
puma(config){admin}# service cookie-persist service-name cookie cookie-name offset offset-len delim delimiter-character
```

2. As admin in config mode, use the following command:

```
puma(config){admin}# build rules
```

Note – Do not run traffic to this service yet, wait for the build to return with completion status.

3. As admin in config mode, use the following command:

```
puma(config){admin}# show build status
```

When the build is completed, the completion message is printed out. Then you can run traffic to this service.

TABLE 4-23 describes the parameters for configuring a service for cookie-based persistence.

TABLE 4-23 Parameters for Configuring a Service for Cookie-Based Persistence

Parameter	Description
<i>service-name</i>	Name of the service entry to be configured for cookie persistence.
<i>cookie</i>	Qualifies the cookie name.
<i>cookie-name</i>	Name of the cookie on which the persistence is enforced.
<i>offset</i>	Qualifies the offset.

TABLE 4-23 Parameters for Configuring a Service for Cookie-Based Persistence

Parameter	Description
<i>offset-len</i>	Number of characters/bytes from the start of the cookie value from which the server name string begins.
<i>delim</i>	Qualifies the delimiter.
<i>delimiter-character</i>	Character used as delimiter to mark the end of the server name string. The only currently available character is a colon (:).

Example

The following example sets the service for cookie-based persistence for the service named `SVC1`, for the cookie named `Car`, the offset length is set for 10 and the delimiter character is a colon.

```
puma(config){admin}# service cookie-persist SVC1 cookie Car offset 10 delim :
```

▼ To Remove Cookie Persistence From a Service

The `remove service cookie-persist` command removes cookie based persistence from a service for a specific cookie name.

- **As admin in config mode, use the following command.**

```
puma(config){admin}# remove service cookie-persist service-name cookie-name
```

Where *service-name* is the name of the service entry and *cookie-name* is the name of the cookie.

Example

The following example removes the service for cookie-based persistence for the service named `SVC1`, for the cookie named `Car`:

```
puma(config){admin}# remove service cookie-persist SVC1 cookie Car
```

▼ To Enable a Load Balancing Service

When a service is created, it is disabled by default. For the service to accept traffic, it must be enabled by invoking the `enable service` command. When a service is enabled, all the load balancing groups it contains get enabled too.

Note – This command fails if the default load balancing group for the service is not configured with at least one active backend server. For an SSL service, this command fails if the service is not configured with at least one active SSL device. For an FTP service, this command fails if the service is not configured with IP persistence.

- As admin in config mode, use the following command:

```
puma(config){admin}# enable service name service-name
```

Where *service-name* is the name of the service to be enabled.

Example

The following example enables the service named SVC1.

```
puma(config){admin}# enable service name SVC1
```

▼ To Disable a Load Balancing Service

The `no enable service` command disables a specified load balancing service. When a service is disabled, all the load balancing groups it contains are disabled. By default all services are disabled upon creation.

- As admin in config mode, use the following command:

```
puma(config){admin}# no enable service name service-name
```

Where *service-name* is the name of the service to be disabled.

▼ To Remove a Load Balancing Service

The `remove service name` command removes one or more load balancing services.

- **As admin, use the following command:**

```
puma(config){admin}# remove service name service-name
```

Where *service-name* is the name of the service to be removed.

Examples

The first example removes one service named `SVC1`.

```
puma(config){admin}# remove service name SVC1
```

The following example removes two services: `SVC1` and `svc2`.

```
puma(config){admin}# remove service name SVC1 svc2
```

Server Configuration

▼ To Enable a Server

The `enable server` command enables a specific back end server on all services or on a specified service. By default the server is enabled on all services.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# enable server {ip-addr | hostname} [service service-name]
```

TABLE 4-24 describes the parameters for enabling a server.

TABLE 4-24 Parameters for Enabling or Disabling a Server

Parameter	Description
<i>ip-addr</i>	Server IP address.
<i>hostname</i>	Server host name.
<i>service</i>	Qualifier for the service name.
<i>service-name</i>	Name of the load balancing service on which the back end server is enabled or disabled.

Examples

The first example enables the server at the IP address of 192.50.50.201 in all services.

```
puma(config){admin}# enable server 192.50.50.201
```

The following example enables the server at the IP address of 192.50.50.201 in the service SVC1.

```
puma(config){admin}# enable server 192.50.50.201 service SVC1
```

▼ To Disable a Server

The `no enable server` command disables a specific back end server on all services or on a specified service. By default, the server is enabled on all services.

Note – If the server is the only active server on any load balancing group, then any subsequent traffic to that server is still sent to the server instead of being dropped.

If the server is the only one in any load balancing group, then this command fails.

- As admin, use the following command:

```
{puma(config){admin}# no enable server {ip-addr | hostname} [service  
service-name]
```


Examples

The first example disables the server at the IP address of 192.50.50.201 in all services.

```
puma(config){admin}# no enable server 192.50.50.201
```

The following example disables the server at the IP address of 192.50.50.201 in the service `SVC1`.

```
puma(config){admin}# no enable server 192.50.50.201 service SVC1
```

▼ To Remove a Server From All Services

- As admin in config mode, use the following command:

```
{puma(config){admin}# remove server {hostname|ip-addr}
```

Where *hostname|ip-addr* is the hostname or IP address of the server to be removed.

Load Balancing Rule Configuration

Creating an IP Load Balancing Rule

The `config ip-rule` command creates a load balancing rule for IP traffic. By default, the rule has low priority.

An IP rule with a high priority gets a higher priority for a given service than IP rules specified with low priority or no priority. When an IP rule is used in conjunction with Layer 7 rules such as HTTP rules in a service, then a high priority puts it at a priority higher than static URLs and a low priority puts it at a priority lower than dynamic URLs. Within multiple IP rules of the same priority class (that is, high or low), the relative priority is determined by the number of bits specified in the IP address and port masks, that is, fewer number of bits masked out results in a higher priority.

Note – The configured rule name must be unique across all types of rules (IP rules, HTTP rules, and so on).

▼ To Create an IP Load Balancing Rule

- As admin in config mode, use the following command:

```
puma(config){admin}# ip-rule name rule ip-addr:port mask ip-addr-mask:port-mask priority
[/high | low]
```

TABLE 4-25 describes the parameters for creating an IP load balancing rule.

TABLE 4-25 Parameters for Creating an IP Load Balancing Rule

Parameter	Description
<i>name</i>	Name of the IP rule created.
<i>rule</i>	Uses the value specified in the <i>ip-addr:port</i> argument.
<i>ip-addr</i>	Source IP address to be matched, for example, 172.88.8.1.
<i>port</i>	Source port to be matched, for example, 21.
<i>mask</i>	Uses the value specified in the <i>ip-addr-mask:port-mask</i> argument.
<i>ip-addr-mask</i>	Source IP address mask to be used for lookup, for example, 255.255.0.0.
<i>port-mask</i>	Source port mask to be used for lookup, for example 0 (for masked) or 1 (specified).
<i>priority</i>	(Optional) Qualifier for the rule priority.
<i>high</i>	(Optional) Specifies the priority of an IP rule as high.
<i>low</i>	(Optional) Specifies the priority of an IP rule as low.

Examples

The first example adds an IP rule named IPRule1 and sets the priority at high.

```
puma(config){admin}# ip-rule IPRule1 rule 172.88.8.1:21 mask 255.255.255.0:1
priority high
```

The following example adds ip-rule IPRule2 and uses the default priority.

```
puma(config){admin}# ip-rule IPRule2 rule 172.88.8.1:3241 mask 255.255.0.0:0
```

Creating an HTTP Load Balancing Rule

The `config http-rule` command creates a load balancing rule for HTTP traffic.

Depending on the rule type, the HTTP rules are assigned different priority classes within a service. Listed in order of decreasing priority, these classes are, static URL, cookie, CGI, and dynamic URLs. Within a particular priority class, individual rules are further prioritized based on the actual rule string, for example, a fully specified rule has a higher priority than a rule with wildcards.

Note – The configured rule name must be unique across all types of rules (IP rules, HTTP rules and such).

Note – The maximum total number of rules is 500. However, if a majority of the rules are complex HTTP rules, there is the possibility that you could exceed the system memory limit before reaching 500 rules. This can also be impacted if there are other large files in the `config` directory.

▼ To Create an HTTP Load Balancing Rule

- As admin in config mode, use the following command:

```
puma(config){admin}# http-rule name {static | dynamic | cgi | cookie} string rule-string
```

TABLE 4-26 describes the parameters for creating an HTTP load balancing rule.

TABLE 4-26 Parameters for Creating an HTTP Load Balancing Rule

Parameter	Description
<i>name</i>	Name of the HTTP rule created.
<i>static</i>	Rule is of the static URL type.
<i>dynamic</i>	Rule is of the dynamic URL type.
<i>cgi</i>	Rule is CGI based.

TABLE 4-26 Parameters for Creating an HTTP Load Balancing Rule (*Continued*)

Parameter	Description
<i>cookie</i>	Rule is cookie-based.
<i>string</i>	Uses the value specified in the <i>rule-string</i> argument.
<i>rule-string</i>	Actual rule string. For example, *.gif can be the rule for a static URL type. The length of the string is restricted to 256 bytes.

Examples

The first example adds an HTTP rule named `HttpR1` of the static URL type with a *.gif *rule-string*.

```
puma(config){admin}# http-rule HttpR1 static string *.gif
```

The following example adds the HTTP rule `HttpCgiR1`, which is CGI-based with `server=server1` as the *rule-string*.

```
puma(config){admin}# http-rule HttpCgiR1 cgi string server=server1
```

The final example adds the HTTP rule `HttpCookieR1`, which is cookie-based with a `server=server2` as the *rule-string*.

```
puma(config){admin}# http-rule HttpCookieR1 cookie string server=server2
```

▼ To Remove a Load Balancing Rule

The `remove rule` command removes one or more a load balancing rules of any type, including IP, HTTP, and others.

If the rule is part of one or more load balancing groups, it cannot be removed.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# remove rule rule-name [rule-name....]
```

Where `rule-name` is the name of the load balancing rule to be removed.

Examples

The following example removes the rules `HttpR1` and `HttpCgiR1` from the system:

```
puma(config){admin}# remove rule HttpR1 HttpCgiR1
```

▼ To Build Load Balancing Rules

The `build rules` command creates a new build for the load balancing rules on the content load balancing blade. This command must be invoked before any modifications made to rules associated with load balancing groups can take effect.

Note – You cannot add, modify, or delete configurations related to the service, lb-group, or rules while the rule building is in progress.

- **As admin in config mode, use the following command:**

```
puma(config){admin}# build rules
```

Note – Even though the CLI for the command returns success immediately, the rules are built by a background task and do not take effect until that task reports successful completion. The status of this background build task can be queried with the `show build status` command.

▼ To Get the Status for the Build for Load Balancing Rules

The `show build status` command displays the status of the current build for the load balancing rules on the content load balancing blade.

- **As any user, use the following command:**

```
puma{user}# show build status
```

▼ To Get the Status for the Last Build for Load Balancing Rules

The `show last build status` command displays the status of the last build for the load balancing rules on content load balancing blade.

- As any user, use the following command:

```
puma{user}# show last build status
```

▼ To Stop the Build for Load Balancing Rules

If a build is in progress, the `no build rules` command stops the current build for the load balancing rules on the content load balancing blade.

- As admin in config mode, use the following command:

```
puma(config){admin}# no build rules
```

Load Balancing Group Configuration and Management

Before a load balancing service can be functional, it must be associated with one or more blade servers, a load balancing scheme, and a load balancing rule. This association is called a load balancing group which is a complete functional unit required for load balancing with the Sun Fire B10n blade.

The available load balancing schemes are as follows:

- Round-robin (`round-robin`)
- Weighted round-robin (`wt-round-robin`)
- Static (`static`)
- Weighted least connections (`wt-least-conn`)
- Response time (`response-time`)

Note – Since the response time load balancing scheme is dependent on application monitoring, a service needs to have its application monitoring parameters configured before a load balancing group can be added to it with the response time load balancing scheme.

▼ To Create a Default Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# service lb-group default service-name server {hostname|ip-addr};port:protocol:weight:active [{hostname|ip-addr};port:protocol:weight:active...] [scheme {round-robin|wt-round-robin|wt-least-conn|response-time|static}]
```

▼ To Create a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# service lb-group name lb-group-name service service-name server {hostname|ip-addr};port:protocol:weight:active [{ip-addr|hostname};port:protocol:weight:active...] rule rule-name [scheme {round-robin|wt-round-robin|wt-least-conn|response-time|static}]
```

This command must be followed by the `build rules` command at the completion of which the new rule added becomes effective.

Whenever a service is created, a default load balancing group is associated with it. So, *default* is not an acceptable value for the group name as it is reserved for the default load balancing group.

An IP rule can be added to a Layer 4 service as well as a Layer 7 service, but HTTP rules can be added only to Layer 7 services.

If the service protocol is configured as UDP, then the only load balancing scheme allowed is static and only IP (Layer 4) rules can be associated with the service.

If the load balancing scheme is static or round-robin, then the weight field is ignored.

If you are configuring a service for UDP, the only load balancing scheme allowed is static.

Whenever a service is created, a default load balancing group is associated with it. See TABLE 4-27.

Note – While you can enter multiple commands in the CLI, no command can be longer than 255 characters. Hence, commands longer than that must be divided into as many commands as necessary to stay within the 255 character limitation.

For example, the following command would fail if you tried to enter it on one line:

```
puma(config){admin}# config service lb-group default lb4rr-8 server
192.168.101.2:6300:tcp:5:1 192.168.101.35:6300:tcp:5:1
192.168.101.4:6300:tcp:5:1 192.168.101.5:6300:tcp:5:1 1
92.168.101.33:6300:tcp:5:1 192.168.101.34:6300:tcp:5:1 scheme wt-round-robin
```

Examples

While the example shows the scheme being defined as weighted round robin, you might prefer some other scheme. Note that weighted round robin can be weighed either by server load or response time.

```
puma(config){admin}# service lb-group name GRP1 service SVC1 server
192.50.50.203:80:tcp:10:1 192.50.50.204:80:tcp:20:1 192.50.50.205:80:tcp:15:0
rule HttpR1 scheme wt-round-robin
```

TABLE 4-27 describes the parameters for creating load balancing groups.

TABLE 4-27 Parameters for Creating Load Balancing Groups

Parameter	Description
name	Qualifier for the LB group name.
<i>lb-group-name</i>	Name of this load balancing group.
service	Uses the value specified in the <i>service-name</i> argument.
<i>service-name</i>	Name of the service.
server	Qualifier for the back end server argument.
<i>ip-addr</i>	IP address of the back end server.
<i>hostname</i>	Host name of the back end server.
<i>port</i>	Port on the back end server where this service can be provided. If specified as 0, it means that this is just a server and not a real service.

TABLE 4-27 Parameters for Creating Load Balancing Groups (Continued)

Parameter	Description
<i>protocol</i>	Corresponding protocol on the back end server. If specified as 0, it means that this is just a server and not a real service.
<i>weight</i>	Weight for this blade server. Valid only if the load balancing scheme used is weighted round robin. Otherwise, specify as 0.
<i>active</i>	Specifies the blade server as active for this load balancing group if the value is 1, standby if the value is 0.
rule	Qualifier for the rule name argument.
<i>rule-name</i>	Name of the load balancing rule to be added to this LB group.
scheme	(Optional) Qualifier for the load balancing scheme.
<i>round-robin</i>	(Optional) Load balancing scheme is round robin.
<i>wt-round-robin</i>	(Optional) Load balancing scheme is weighted round robin.
<i>wt-least-conn</i>	(Optional) Load balancing scheme is weighted least connection.
<i>response-time</i>	(Optional) Load balancing scheme is response time.
<i>static</i>	(Optional) Load balancing scheme is static load balanced, where the server is chosen by a hash function. Used when the service is configured for UDP.

Note – Port NAT is not supported at this time. So the *port* and *protocol* field values are ignored.

▼ To Add Rules to a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# service lb-group rule service-name:lb-group-name rule rule-name [rule-name.....]
```

The `service lb-group rule` command adds one or more rules to a load balancing group. When a request matches this service, it is matched against all the rules linked with the service and if any rule matches the request, then it is load balanced to the servers configured for this load balancing group.

This command should be followed by the `build` rules at the completion of which the new rule(s) added becomes effective.

A rule (with a given content) can be added to a service only once, that is, it can appear only once in at most one load balancing group for a given service. Addition of duplicate rules to a service fails.

An IP rule can be added to a Layer 4 service as well as a Layer 7 service, but HTTP rules can be added only to Layer 7 services.

Note – This command cannot be used to add rules to the default group.

TABLE 4-28 describes the parameters for adding rules to a load balancing group.

TABLE 4-28 Parameters for Adding Rules to a Load Balancing Group

Parameter	Description
<i>service-name</i>	Name of the load balancing service to which this LB group belongs.
<i>lb-group-name</i>	Name of the load balancing group to which one or more rules are being added.
<i>rule</i>	Uses the value specified in the <i>rule-name</i> argument.
<i>rule-name</i>	Name of the rule to be added to this LB group.

Examples

The first example adds an HTTP cookie rule to SVC1 and GRP1.

```
puma(config){admin}# service lb-group rule SVC1:GRP1 rule HttpCookieR1
```

The following example adds a CGI rule to SVC1.

```
puma(config){admin}# service lb-group rule SVC1:GRP1 rule HttpCgiR1
```

▼ To Remove Rules From a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# remove service lb-group rule service-name rule rule-name [rule-name.....]
```

The `remove service lb-group rule` command removes one or more rules from a load balancing group.

This command should be followed by a `build rules` command after the completion of which the rule removal becomes effective.

If the rule being removed from the load balancing group is the last rule present and the service is configured for Layer 7 load balancing, then this command fails.

Example

```
puma(config){admin}# remove service lb-group rule SVC1 rule HttpCgiR1
```

▼ To Add Servers to a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# service lb-group server service-name:lb-group-name server {ip-addr | hostname};port:protocol:weight:active [{ip-addr | hostname};port:protocol:weight:active...]
```

The `service lb-group server` command adds one or more servers to a load balancing group.

If the load balancing scheme is weighted round robin and the weight is specified as 0 for a server, then the default weight is 1.

TABLE 4-29 describes the parameters for adding servers to a load balancing group.

TABLE 4-29 Parameters for Adding Servers to a LB Group

Parameter	Description
<i>service-name</i>	Name of the load balancing service to which this LB group belongs.
<i>lb-group-name</i>	Name of the load balancing group to which one or more server-port pairs are being added.
<i>server</i>	Qualifier for the blade server argument.
<i>ip-addr</i>	Server IP address.
<i>hostname</i>	Server host name.
<i>port</i>	Port on the back end server where this service can be provided.
<i>protocol</i>	Corresponding protocol on the blade server.
<i>weight</i>	Weight for this blade server. Valid only if the load balancing scheme used is weighted round robin. Otherwise, this is ignored.
<i>active</i>	Specifies the back end server as active for this load balancing service group if the value is 1, standby if the value is 0.

Note – Port NAT is not supported at this time. So the *port* and *protocol* field values are ignored.

Example

```
puma(config){admin}# service lb-group server SVC1:default server  
192.50.50.210:80:10:1
```

▼ To Remove Servers From a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# remove service lb-group server service-name:lb-group-name server  
{ip-addr | hostname}:port:protocol [{ip-addr | hostname}:port:protocol...]
```

The `remove service lb-group server` command removes one or more servers from a load balancing group.

If the server being removed from the load balancing group is the last one present, then this command fails.

TABLE 4-30 describes the parameters for removing servers from a load balancing group.

TABLE 4-30 Parameters for Removing Servers to a LB Group

Parameter	Description
<i>service-name</i>	Name of the load balancing service to which this LB group belongs.
<i>server</i>	Qualifier for the back end server argument.
<i>ip-addr</i>	Server IP address.
<i>hostname</i>	Server host name.
<i>port</i>	Port on the back end server where this service can be provided.
<i>protocol</i>	Corresponding protocol on the back end server.

Examples

The following example removes the service SVC1 from lb-group server with the IP address of 192.50.50.210 on port 80.

```
puma(config){admin}# remove service lb-group server SVC1:default server 192.50.50.210:80
```

▼ To Set Servers for a Load Balancing Group as Active or Standby

- As admin in config mode, use the following command:

```
puma(config){admin}# modify service lb-group server service-name:lb-group-name server {hostname | ip-addr}:port:protocol [{hostname | ip-addr}:port:protocol...] mode {active|standby}
```

This command sets one or more servers for a load balancing group as active or standby.

Note – A load balancing group must have at least one active server.

TABLE 4-31 describes the parameters for this command.

TABLE 4-31 Parameters for Setting Servers to Active or Standby

Parameter	Description
<i>server</i>	Qualifier for the <i>service-name:lb-group-name</i> argument.
<i>service-name:lb-group-name</i>	Name of the load balancing service to which this load balancing group belongs, and the name of the load balancing group, separated by a colon.
<i>server</i>	Qualifier for the server or service.
<i>hostname</i>	Server host name.
<i>ip-addr</i>	Server IP address.
<i>port</i>	Port number on the server where this service is being offered.
<i>protocol</i>	Corresponding protocol on the back end server.
<i>mode</i>	Qualifier for the server mode.
<i>active</i>	Sets server or service as active for this LB group.
<i>standby</i>	Sets server or service as standby for this LB group.

Note – Port NAT is not supported at this time. So the *port* and *protocol* field values are ignored.

Example

The following example, modifies service SVC1 on the server with the IP address of 192.50.50.210 at port 80 and places it in standby mode.

```
puma(config){admin}# modify service lb-group server SVC1:default server  
192.50.50.210:80:tcp mode standby
```

▼ To Modify the Load Balancing Scheme of a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# modify service lb-group scheme service-name:lb-group-name scheme
{round-robin | wt-round-robin | wt-least-conn | response-time | static}
```

TABLE 4-32 describes the parameters for this command.

TABLE 4-32 Parameters for Modifying a Load Balancing Group Scheme

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<i>lb-group-name</i>	Name of the load balancing group.
<i>scheme</i>	Qualifier for the load balancing algorithm argument.
<i>round-robin</i>	Configures the load balancing scheme to be round-robin.
<i>wt-round-robin</i>	Configures the load balancing scheme to be weighted round-robin.
<i>wt-least-conn</i>	Configures the load balancing scheme to be weighted least connections.
<i>response-time</i>	Configures the load balancing scheme to be response time.
<i>static</i>	Configures the load balancing scheme to be static.

▼ To Modify the Weight of a Server in a Load Balancing Group

- As admin in config mode, use the following command:

```
puma(config){admin}# modify service lb-group weight service-name:lb-group-name server
{hostname | ip-addr}:port:proto:wgt [[hostname | ip-addr]:port:proto:wgt...]
```

TABLE 4-32 describes the parameters for this command.

TABLE 4-33 Parameters for Modifying a Load Balancing Group Scheme

Parameter	Description
<i>service-name</i>	Name of the load balancing service.
<i>lb-group-name</i>	Name of the load balancing group.
<i>server</i>	Qualifier for the server(s) argument.
<i>hostname</i>	Hostname of the server being modified.
<i>ip-addr</i>	IP address of the server being modified.
<i>port</i>	Port of the server being modified.
<i>proto</i>	Protocol of the server being modified.
<i>wgt</i>	New weight value for the server being modified.

▼ To Remove Load Balancing Groups

- As admin in config mode, use the following command:

```
puma(config){admin}# remove service lb-group name service-name:lb-group-name [service-name:lb-group-name]
```

The `remove service lb-group` command removes one or more load balancing groups from a service.

The default load balancing group cannot be removed by this command.

This command should always be followed by a `build rules` command before the removal of the rules contained in this load balancing group is effective.

Examples

The following example removes the load balancing group GRP1 from the service SVC1.

```
puma(config){admin}# remove service lb-group name SVC1:GRP1
```

Load Balancing Configuration Listings

The Sun Fire B10n blade enables you to list load balancing configurations by service, rule, group, or servers.

▼ To List Load Balancing Services

The `show service` command lists the configurations of a particular service if specified. Otherwise this command lists the configurations of all the load balancing services that have been created. By default, all services are listed.

- **As any user, use the following command:**

```
B10n {user}# show service service-name
```

Where *service-name* is the name of the service to be retrieved.

Returns

Table of services listed by name with the values of the following basic parameters listed against each service, or the configurations of a single service if specified:

- Primary service point IP address and port
- Service protocol (TCP/UDP)
- Layer at which the service is load balanced (Layer 4 or Layer 7)
- Status of the service (enabled/disabled)
- List of default servers and default load balancing scheme
- List of service association names linked to the service
- VLAN tag for data traffic if configured
- Persistence (IP persistence and service point tracking) configurations, if any
- SSL configurations if any
- list of service points (VIP and port) with the SSL support (whether SSL enabled or not) and the binding interface listed for each service point
- DoS avoidance status (enabled/disabled)

▼ To List Load Balancing Rules

The `show rule` command lists a particular load balancing rule if specified. Otherwise the command lists all the load balancing rules that have been created so far. By default, all rules are listed.

- As any user, use the following command:

```
puma{user}# show rule [rule-name]
```

Where `rule-name` is the name of the rule to be retrieved.

Returns

A list of load balancing rule names with their respective type and content specified (or the type and content of a single rule if specified). If a particular rule name is specified, then all the services and the corresponding load balancing group in which the rule is included in each service is also listed.

Examples

To show all load balancing rule names and their respective type and content, use the following example:

```
puma{user}# show rule
```

To see specific rule type and content, specify the rule name:

```
puma{user}# show rule HttpCookieR1
```

▼ To List Load Balancing Groups

- As any user, use the following command:

```
puma{user}# show service-lb-group service-name [lb-group-name]
```

Where:

service-name is the name of the load balancing service to which this load balancing group belongs.

lb-group is the name of the load balancing group entry to be retrieved.

The `show service-lb-group` command lists the information about a particular load balancing group if specified. Otherwise this command lists the information about all the load balancing groups for a given service.

Returns

Table containing all the relevant load balancing groups. Each group has a row containing the following information:

- Load balancing group name
- Name of the service to which it belongs
- Names and actual rule strings of all the rules linked with it
- Load balancing scheme used
- All the servers and services associated with it (with the weight specified against each server if the load balancing scheme is weighted round robin)

If a load balancing group is specified, then all this information is listed for that particular group.

Examples

```
puma{user}# show service-lb-group SVC1
```

```
puma{user}# show service-lb-group SVC1 default
```

▼ To List Servers

The `show server` command lists a particular server if specified. Otherwise, all the servers in the system that are included in one or more load balancing service associations for all the services configured on the content load balancing blade are listed

- **As any user, use the following command:**

```
puma{user}# show server [hostname|ip-addr]
```

Where `hostname` is the name and `ip-addr` is the IP address of the server to be retrieved.

Returns

A list of server names with the following specified for each server:

- server IP address
- enabled/disabled
- up/down

If a particular server name is specified, then all the services and the corresponding load balancing groups in which the server is included in each service are also listed.

Configuring the System

The Sun Fire B10n blade can be loaded with three different images and booted. The three images are `image 1`, `image 2`, and `diag`.

The image can be upgraded interactively or non interactively.

▼ To Configure the Image for the Next Reboot

1. **As admin in config mode, use the following command:**

```
puma(config){admin}# boot image {1|2|diag}
```

Options for the `boot image` are `1`, `2`, or `diag`.

Example

This example sets the diagnostics image to be used for the next reboot.

1. Set the diagnostics image to be used for the next reboot:

```
puma(config){admin}# boot image diag
```

2. To make this the permanent setting, use the commit command:

```
puma(config){admin}# commit
```

3. Reboot the system:

```
puma(config){admin}# reboot
```

▼ To Download a New Boot Image Over the Network

1. As admin, use the following command:

```
puma{admin}# update image {hostname|ip-addr} file filename image  
{1|2|diag}
```

This command downloads a new boot image over the network and writes it into flash PROM. The new image takes effect after a system reboot. This command is available in interactive and noninteractive modes.

Example

The following example updates image 2 using the file pkgname command from the remote server at IP address 192.50.50.201:

1. As admin, use the following command:

```
puma{admin}# update image 192.50.50.201 file pkgname image 2
```

2. Reboot the system:

```
puma{admin}# reboot
```

▼ To Configure the Diagnostics Level

The `diag level` command configures the diagnostics level and also the level of verbosity of the diagnostics.

- As admin in config mode, use the following command:

```
puma(config){admin}# diag level {0|1|2} verbose {0|1|2}
```

TABLE 4-34 describes the parameters for configuring the diagnostic and verbosity level.

TABLE 4-34 Parameters for Configuring the Diagnostic and Verbosity Level

Parameter	Description
0	Minimum level of diagnostics and verbosity.
1	Intermediate level of diagnostics and verbosity.
2	Maximum level of diagnostics and verbosity.

By default, the diagnostics level is 0 and the verbosity level is 0.

These values are used whenever the system boots with the diag image.

Examples

The following example configures the diagnostic level as 1 and the verbosity as 2:

```
puma(config){admin}# diag level 1 verbose 2
```

▼ To Configure the Debug Level for Specific Modules

The `debug module` command configures the debug level for a specified module in the system.

- As admin in config mode, use the following command:

```
puma(config){admin}# debug module module-name level {0-5}
```

TABLE 4-35 describes the parameters for configuring the debug level for specified modules:

TABLE 4-35 Parameters for Configuring the Debug Level for Specific Modules

Parameter	Description
<i>module-name</i>	Name of the module.
0-5	Allowed range of values for the debug level. A debug level of 0 means debug is turned off while a level of 5 means all debug messages are activated. The default level is 0.

TABLE 4-36 describes the module names to use for configuring the debug level:

TABLE 4-36 Module Names to Use for Configuring the Debug Level

Module Name	Description
lb	Load balancing module.
network	Networking module.
failover	Failover module.
mgmt	Management module.
sys	System-level modules.
npu-if	Module interfacing with the NPU.
class-if	Module interfacing with the Classifier.

Example

The following example configures the load balancing module at level 2.

```
puma(config){admin}# debug module lb level 2
```

▼ To Shutdown the System

The `shutdown` command does a graceful shutdown of the system. This command is available in both interactive and non-interactive mode. By default, this command is in the interactive mode and asks for confirmation before shutting down the system.

- As admin in config mode, use the following command:

```
puma(config){admin}# shutdown [force]
```

Where `force` forces the shutdown without asking for confirmation.

Examples

The first example shuts down the system using the interactive mode.:

```
puma(config){admin}# shutdown
```

The following example shuts down the system using the non-interactive mode:

```
puma(config){admin}# shutdown force
```

▼ To Reboot the System

The `reboot` command resets the system. It is available in both interactive and non-interactive mode. By default, this command is in the interactive mode and asks the user for confirmation before rebooting the system.

- As admin, use the following command:

```
puma(config){admin}# reboot [force]
```

Where `force` forces the reboot without asking for confirmation. This command checks if there is a difference between the running configuration and the saved configuration. This command gives a warning for unsaved configurations before rebooting.

Examples

The first example reboots the system using the interactive mode.:

```
puma(config){admin}# reboot
```

The following example reboots the system using the non-interactive mode:

```
puma(config){admin}# reboot force
```

▼ To Show the Date and Time

Shows the current system date and time.

- **As any user, use the following command:**

```
puma{user}# show date
```

▼ To Show the System Settings on the B10n

The `show system` command shows the current system settings. It gives information about the current image, its version, the current configuration file in flash being used, and so on.

- **As any user, use the following command:**

```
puma{user}# show system
```

▼ To Show the System Uptime

The `show uptime` command shows the uptime for the system.

- **As any user, use the following command:**

```
puma{user}# show uptime
```

▼ To Show All of the Blade Configurations on the B10n

The `show configuration` command lists all of the blade configurations as the collective output from commands: `show network`, `show service`, `show server`, `show rule` and `show vip`.

- As any user, use the following command:

```
puma{user}# show configuration
```

▼ To Compare the Running Configuration With the Saved Configuration

The `show compare-config` command compares the configuration in running memory with its correspondent configuration saved in the Flash File System. This helps you determine if the configuration has been changed and if the need to save the configuration is required.

- As any user, use the following command:

```
puma{user}# show compare-config
```

▼ To Show the Configuration in Running Memory

The `show running-config` command displays the configuration that is in the running memory.

- As any user, use the following command:

```
puma{user}# show running-config
```

▼ To Show the Configuration Saved in Flash Memory

The `show saved-config` command shows the configuration saved in the Flash File System with the option of 1 or 2. 1 indicates `config-1` and 2 indicates `config-2`.

- As any user, use the following command:

```
puma{user}# show saved-config
```

▼ To Show Memory

The `dump memory` command dumps the system memory to the screen. By default 32 bytes are dumped starting from the specified memory location.

- As any user, use the following command:

```
puma{user}# dump memory addr [size]
```

TABLE 4-37 describes the parameters for the `dump memory` command:

TABLE 4-37 Parameters for the `dump memory` Command

Parameter	Description
<code>addr</code>	Address on the system memory to dump from (in hex without the leading '0x').
<code>size</code>	Number of bytes to dump.

Examples

The first example dumps memory from address `56789abc` using the default size.

```
puma{user1}# dump memory 56789abc
```

The following example dumps memory from address 56789abc, but specifies the size of 8 bytes:.

```
puma{user1}# dump memory 56789abc 8
```

▼ To Show Modules

The `dump module` command dumps the information regarding a specific module to the screen.

- As any user, use the following command:

```
puma{user}# dump module module-name [index]
```

TABLE 4-38 describes the parameters for the `dump module` command:

TABLE 4-38 Parameters for the `dump module` Command

Parameter	Description
<i>module-name</i>	Name of the module.
<i>index</i>	(Optional) Index of the sub-module.

TABLE 4-39 describes the valid module names to use with `dump module` command:

TABLE 4-39 Module Names to Use with the `dump module` Command

Module Name	Description
help	Displays all the module names available.
npu	Displays the NPU hardware and driver information.
classifier	Displays the classifier device registers.
sysctl	Displays the system controller registers.
vxworks	Displays the <code>vxworks</code> tasks information.
task	Displays the registers and stack trace for <code>vxworks</code> tasks.
network	Displays the <code>vxworks</code> network tables/statistics.

TABLE 4-39 Module Names to Use with the `dump module` Command

Module Name	Description
<code>failover</code>	Displays failover information.
<code>debug</code>	Displays debug information.
<code>stats</code>	Displays the statistics.

The following table lists the modules in the NPU by index:

TABLE 4-40 Modules in the NPU Listed by Index

Index	Sub-Module Name
0	Lists the sub modules available by index
1	BAD
2	MID
3	HIF
4	OM
5	PIM A
6	PIM B
7	PIM C
8	PIM D
9	POM A
10	POM B
11	POM C
12	POM D
13	PPE A
14	PPE B
15	PPE C
16	PPE D
17	SB A
18	SB B
19	SB C
20	SB D
21	CS
22	LAB

TABLE 4-40 Modules in the NPU Listed by Index *(Continued)*

Index	Sub-Module Name
23	GMAC A
24	GMAC B
25	SPI3 IF
26	HAL
27	Server Table Allocator
28	LB Group Allocator
29	Server Entry Allocator
30	MEMC
200	Displays the Registers and Memory for all PPEs
201	Displays the Registers and Memory for PPE A
202	Displays the Registers and Memory for PPE B
203	Displays the Registers and Memory for PPE C
204	Displays the Registers and Memory for PPE D
255	Displays NPU statistics per PPE (in hexadecimal)

Examples

The following example displays the Order Manager (OM) module in the NPU.

```
puma{user1}# dump module npu 4
```

The following table lists the tasks available for display by index in the `dump module task` command.

TABLE 4-41 Tasks Listed by Index for the `dump module task` Command

Index	Available Tasks
0	Lists the tasks available by index.
1	Displays <code>tNetTask</code> .
2	Displays <code>tPerUpdates</code> .
3	Displays <code>tClbRx</code> .
4	Displays <code>tClbRetx</code> .
5	Displays <code>tSrvrMon</code> .

TABLE 4-41 Tasks Listed by Index for the `dump module task` Command (Continued)

Index	Available Tasks
6	Displays <code>tCleanup</code> .
7	Displays <code>tSlowCleanup</code> .
8	Displays <code>tBuildRules</code> .
9	Displays <code>tFoSync</code> .
10	Displays <code>tFoMonitor</code> .
11	Displays <code>tRpcSvc</code> .
12	Displays <code>TELNETD</code> .
13	Displays <code>tExcTask</code> .
14	Displays <code>tLogTask</code> .
15	Displays <code>CONSOLE</code> .

The following table lists the VxWorks network tables/statistics for display by index in the “`dump module network`” command.

TABLE 4-42 VxWorks Network Statistics/Tables by Index

Index	Network Statistics
0	Lists the statistics/tables available by index
1	Displays the IP stats.
2	Displays net pool stats.
3	Displays stack data pool stats.
4	Displays stack system pool stats.
5	Displays mbuf stats.
6	Displays the ARP table.
7	Displays routing stats.
8	Displays routes.
9	Displays the host table.
10	Displays active connections.
11	Displays the interface information.

The following table lists the failover sub-modules available by index in the `dump module failover` command.

TABLE 4-43 Failover Sub-modules by Index

Index	Failover Sub-modules
0	Lists the sub-modules available by index.
1	Displays the failover monitoring module information.
2	Lists the <code>ramdisk</code> directory.
3	Displays the failover state information.

The following table lists the statistics available by index in the `dump module stats` command.

TABLE 4-44 Statistics Listed by Index for the `dump module stats` Command

Index	Statistics
0	Lists the statistics available by index.
1	Displays the NPU Statistics.
2	Displays the Host Statistics.

▼ To Export a File to a Remote Host

The `export file` command exports a file to another machine. After you specify the user name for logging into the remote host, the content load balancing blade responds to this command with a prompt for the password.

This interactive command prompts you for the hostname/IP address of the remote host to export the file to, the file to export, the path on the remote host to export the file to, the username, and the password.

- **As admin, use the following command:**

```
puma{admin}# export file
```

Note – You can use this command to export a configuration file to a remote host.

▼ To Import a File From a Remote Host

The `import file` command imports a file from another machine. After you specify the user name for logging into the remote host, the content load balancing blade responds to this command with a prompt for the password.

This interactive command prompts you for the hostname/IP address of the remote host to import the file from, the file to import, the path on the remote host to import the file from, the username, and the password.

- As admin, use the following command:

```
puma{admin}# import file
```

Note – You can use this command to import a configuration file to a remote host.

▼ To Commit the Current Configuration

- As admin in config mode, use the following command:

```
puma(config){admin}# commit [force]
```

This command saves all the configuration writes into nonvolatile memory so that the writes can be recovered upon a restart. This is an interactive command and it prompts you for confirmation. To bypass the interactive mode, use the `force` option.

▼ To Show the Current Configuration in Flash

- As any user, use the following command:

```
puma{user}# dump config
```

This command displays all the configurations saved in the flash memory.

▼ To Save this Current Configuration in Flash

1. Copy the config that gets printed to the screen.
2. Paste the config into a file to keep it for future use.

▼ To Remove the Current Configuration in Flash

The `erase config-files` command erases the current configuration in flash. By default, the command is interactive and asks for confirmation before removing the configuration in flash memory.

A maximum of two configuration files can be in flash memory. The configuration file that will be removed by the `erase config-files` command is the current configuration file, which you can obtain with the `show system` command.

- As admin in config mode, use the following command:

```
puma(config){admin}# erase config-files [force]
```

Where `force` forces the removal of the current configuration file.

Example

The following example forces the removal of the current configuration file:

```
puma(config){admin}# erase config-files force
```

▼ To Specify the Configuration in Flash to Use After a System Reboot

There can be two configuration files in flash. The `boot config` command specifies which configuration file to use the next time the load balancing blade comes up after a system reboot.

- As admin in config mode, use the following command:

```
puma(config){admin}# boot config {1|2}
```

Where 1 is configuration 1 and 2 is configuration 2.

Examples

The following example specifies that configuration file 2 be used when the system reboots:

```
puma(config){admin}# boot config 2
```

Flash File System Commands

▼ To Check or Repair the Flash File System

The `chkdsk {check|repair}` command verifies that the Flash File System is in good condition. The `check` option determines the condition of the Flash File System. The `repair` option fixes problems found in checking process.

- As admin, use the following command:

```
puma{user}# chkdsk {check|repair}
```

▼ To Output a File to the Screen

- As admin, use the following command:

```
puma{admin}# cat filename
```

The `cat` command outputs a file in the flash file system to the screen.

▼ To Change the Current Directory

- As any user, use the following command:

```
puma{user}# cd new-directory
```

▼ To Copy a File

- As any user, use the following command:

```
puma{user}# cp src-file dst-file
```

▼ To Rename a File

- As admin in non-config mode, use the following command:

```
puma{admin}# mv old-file new-file
```

▼ To Delete a File

- As admin in non-config mode, use the following command:

```
puma{admin}# rm filename
```

This command allows the star (*) wildcard as its argument, for example: *. * or *.x or x.* or *.

▼ To Create a New Directory

- As admin in non-config mode, use the following command:

```
puma{admin}# mkdir dir-name
```

▼ To Remove a Directory

- As admin, use the following command:

```
puma{admin}# rmdir dir-name
```

This command can remove the directory recursively if the top level directory has children directories. However, this command cannot remove any system level directories such as the config directory and boot image directory.

▼ To List Files

- As any user, use the following command:

```
puma{user}# ls
```

▼ To Print the Current Working Directory

- As any user, use the following command:

```
puma{user}# pwd
```

▼ To Compress All Files in a Directory

- As any user, use the following command:

```
puma{user}# tar tar-filename dir-name
```

Where *tar-filename* is the name of the tar file created and *dir-name* is relative path to the file or directory being compressed.

▼ To Uncompress Files

- As any user, use the following command:

```
puma{user}# untar tar-filename
```

Where, *tar-filename* is the name of the compressed file.

▼ To Display Contents of a Compressed File

- As any user, use the following command:

```
puma{user}# tarinfo tar-filename
```

Where, *tar-filename* is the name of the compressed file whose contents you want to display.

Other Useful Commands

▼ To Clear the Screen

- As any user, enter the following command

```
puma{user}# clear
```

▼ To Create an Alias for Any Command

- As any user, use the following command:

```
puma{user}# alias alias-name <command for which alias is created>
```

▼ To Send a Message to All Logged-on Users

- As any user, use the following command:

```
puma{user}# broadcast "mesg"
```

Where *mesg* is the message string to be broadcast. The string must be surrounded by double quotes.

Use this command to send a message to all the users logged into the Sun Fire B10n blade.

▼ To Echo a String on the Screen

- As any user, use the following command

```
puma{user}# echo string
```

▼ To View the Command-Line Interface Tree

- As any user, use the following command:

```
puma{user}# tree
```

Returns the command-line interface tree.

```
sc0> console s1
[connected with input enabled]
puma{admin}# tree
|
+---alias
|
+---broadcast
|
+---cat
|
+---cd
|
+---chkdsk
|
```

```

+---clear
|
+---commit
|
+---config
|
|   +---boot
|   |
|   |   +---config
|   |   |
|   |   +---image
|   |
|   +---build
|   |
|   |   +---rules
|   |
|   +---data
|   |
|   |   +---vlan
|   |
|   +---debug
|   |
|   +---default
|   |
|   |   +---gateway
|   |   |
|   |   +---hostname
|   |   |
|   |   +---qos
|   |   |
|   |   +---tcp-dos-params
|   |   |
|   |   +---tcp-handoff-params
|   |   |
|   |   +---tcp-params
|   |
|   +---diag
|   |
|   +---dns
|   |
|   |   +---server
|   |   |
|   |   +---suffix
|   |
|   +---enable
|   |
|   |   +---failover-monitor
|   |   |
|   |   +---path-failover

```



```

+---management
|
|   +---vlan
|
+---modify
|
|   +---service
|       |
|       +---lb-group
|           |
|           +---server
|
|       +---ssl
|           |
|           +---mode
|
+---path-failover
|
|   +---target
|
+---path-failover-monitor
|
+---remove
|
|   +---dns
|
|   +---failover-config
|
|   +---path-failover
|
|   +---rule
|
|   +---service
|       |
|       +---cookie-persist
|
|       +---lb-group
|           |
|           +---name
|
|           +---rule
|
|           +---server
|
|       +---name
|
|       +---point
|
|       +---ssl

```

```

|
| +---tracking
|
| +---ssl
|   |
|   +---if
|   |
|   +---name
|   |
|   +---port-pair
|
+---server-monitor
|
+---service
|   |
|   +---cookie-persist
|   |
|   +---ip-persist
|   |
|   +---lb-group
|   |   |
|   |   +---default
|   |   |
|   |   +---name
|   |   |
|   |   +---rule
|   |   |
|   |   +---server
|   |
|   +---name
|   |
|   +---point
|   |
|   +---qos
|   |
|   +---ssl
|   |
|   +---tcp-dos-params
|   |
|   +---tcp-handoff-params
|   |
|   +---tcp-params
|   |
|   +---tracking
|   |
|   +---vlan
|
+---ssl
|

```

```
|
|
|     +---if
|     |
|     +---name
|     |
|     +---port-pair
|
|     +---vip-broadcast
|     |
|     +---vip-netmask
| +---cp
+---dump
|
|     +---config
|     |
|     +---memory
|     |
|     +---module
|
+---echo
|
+---exec
|
+---exit
|
+---export
|   |
|   +---file
|
+---help
|
+---history
|
+---import
|   |
|   +---file
|
+---login
|
+---logout
|
+---ls
|
+---mkdir
|
+---mv
|
+---ping
|
```

```
+---pwd
|
+---reboot
|
+---rm
|
+---rmdir
|
+---show
|   |
|   +---arp
|   |
|   +---build
|   |   |
|   |   +---status
|   |
|   +---network
|   |
|   +---rule
|   |
|   +---running-config
|   |
|   +---saved-config
|   |
|   +---server
|   |
|   +---service
|   |
|   +---service-lb-group
|   |
|   +---ssl
|   |
|   +---system
|   |
|   +---uptime
|   |
|   +---user
|   |
|   +---vip
|   |
|   +---vlan
|
+---stty
|   |
|   +---hardwrap
|   |
|   +---status
|
+---tar
```

```
|
+---tarinfo
|
+---tree
|
+---untar
|
+---update
|   |
|   +---image
|
+---user
|   |
|   +---access
|   |
|   +---add
|   |
|   +---delete
|   |
|   +---password
|   |
|   +---show
|
+---who
|
+---whoami
|
+---write
```

▼ To Print the History of All Executed Commands

- As any user, use the following command:

```
puma{user}# history
```

Prints out all the commands executed till now in this session.

▼ To Get Help for CLI Commands

- As any user, use the following command:

```
puma{user}# help command
```

Using the `help` command alone gets help on all commands. Entering a specific command, such as `set service lb-group server`, returns help for that specific command.

You can also get help by entering the command name and a question mark (?), for example:

```
puma{user}# service ?
```

▼ To Logout

- As any user, use the following command:

```
puma{user}# logout
```

Logs out from the current console session.

▼ To Exit From a Script

- As any user, use the following command:

```
puma{user}# exit
```

▼ To Retrieve the Current User Information

- As any user, use the following command:

```
puma{user}# whoami
```

▼ To Retrieve Information About All Users

- As any user, use the following command:

```
puma{user}# who
```

▼ To Display Console Settings

- As any user, use the following command:

```
puma{user}# stty status
```

Use this command to display the console settings.

▼ To Turn On Hardwrap on the Console

- As any user, use the following command:

```
puma{user}# stty hardwrap
```


Configuring Failover

This chapter describes how to configure path failover and blade failover for the Sun Fire B10n blade. The following sections are included:

- “Configuring Path Failover” on page 142
- “Configuring Blade Failover” on page 148

The Sun Fire B10n blade offers two levels of failover: path failover and blade failover.

- **Path Failover:**

For failover within a blade, an alternate network interface path is used when the current active network path fails.

- **Blade Failover:**

For failover between Sun Fire B10n blades deployed in pairs, the standby blade takes over when it fails to read the active blade or failover is manually forced. Only one of these two blades is active at any given time but each blade is aware of the other by exchanging the failover state information.

When a failover is configured and enabled, monitoring packets containing the failover state data will be sent to the peer periodically. One usage of this state data is for the blade to negotiate and determine its role as either the active or standby blade. One other usage is for the standby blade to detect the active failure and initiate a failover.

If the standby blade does not receive failover state information from the active blade after a configurable interval and number of retries (`max-try`), the standby takes over and becomes active. The blade failover does not support stateful failover. This implies that any open connections will be terminated if the active blade fails.

To force the standby blade to be the active blade, a failover force command may be executed to switch roles.

Configuring Path Failover

▼ To Configure IPMP On a Sun Fire B100 When Using Interfaces on a Sun Fire B100 as the Target Paths

The B100 requires four IP addresses: one for each interface (`ce0` and `ce1`) plus one test address for each interface. The test addresses are used to perform a ping. If the B100 doesn't receive a reply from a ping on the test address associated with one interface, it knows that the interface has failed and it directs all network traffic for either interface over the valid one.

The following steps are an example; replace IP addresses and hostnames as appropriate.

1. Access the console of the B100 from the system controller.

```
sc> console sn
```

Where `n` is the number of the slot containing the B100 you want to log into.

2. Choose one IP address for your `ce0` interface and one IP address for your `ce1` interface. Also choose one IP address for your test `ce0` interface and one IP address for your test `ce1` interface. Add these addresses to the `/etc/hosts` file.

```
# /etc/hosts on the B100

data-ce0  192.168.101.240    # Data Address for ce0
data-ce1  192.168.101.117    # Data Address for ce1
test-ce0  192.168.101.241    # Test Address for ce0
test-ce1  192.168.101.118    # Test Address for ce1
```

3. Create the interfaces `ce0` and `ce1`.

```
# ifconfig ce0 plumb
# ifconfig ce1 plumb
```

4. Create an IPMP group named `ipmp-group` containing `ce0` and `ce1` interfaces.

```
# ifconfig ce0 group ipmp-group
# ifconfig ce1 group ipmp-group
```

5. Create an address on `ce0` and `ce1` for data transmission and mark it to failover if an interface failure is detected.

```
# ifconfig ce0 data-ce0 ipmp-group netmask + broadcast + failover up
# ifconfig ce1 data-ce1 ipmp-group netmask + broadcast + failover up
```

6. Configure a test address for `ce0` and `ce1`.

```
# ifconfig ce0 addif test-ce0 netmask + broadcast + -failover deprecated up
# ifconfig ce1 addif test-ce1 netmask + broadcast + -failover deprecated up
```

7. To enable the new interfaces configuration to survive a reboot, create both a `/etc/hostname.ce0` and `/etc/hostname.ce1` file.

`/etc/hostname.ce1:`

```
data-ce0 netmask + broadcast + group ipmp-group up \
addif test-ce0 deprecated -failover netmask + broadcast + up
```

`/etc/hostname.ce1:`

```
data-ce1 netmask + broadcast + group ipmp-group up \
addif test-ce1 deprecated -failover netmask + broadcast + up
```

8. Check that `ce0`, `ce1`, `ce0:1`, and `ce1:1` are created correctly with the `ifconfig` command. `ce0:1` and `ce1:1` are the two test interfaces.

```
# ifconfig -a
```

Please refer to the *Sun Fire B1600 Blade System Chassis Software Setup Guide* if you need more detailed information. This guide also describes how to configure IPMP on a Sun Fire B100 with VLANs.

▼ To Add a Path Failover Target Address to an Interface

- Use the following command:

```
puma{admin}# config path-failover target interface {0|1} {hostname|IP_address}
```

Both interface 0 and interface 1 must be configured for path failover. So, this command must be executed for both interface 0 and for interface 1.

Example

```
puma{admin}# config path-failover target interface 0 192.168.101.240
```

```
puma{admin}# config path-failover target interface 1 192.168.101.117
```

▼ To Remove a Path Failover Target Address on an Interface

- Use the following command:

```
puma{admin}# config remove path-failover interface {0|1}
```

Example

```
puma{admin}# config remove path-failover interface 0
```

▼ To Enable Path Failover Monitoring

- Use the following command:

```
puma{admin}# config enable path-failover
```

Example

```
puma{admin}# config enable path-failover
```

▼ To Disable Path Failover Monitoring

- Use the following command:

```
puma{admin}# config no enable path-failover
```

Example

```
puma{admin}# config no enable path-failover
```

▼ To Configure Path Failover Monitoring Parameters

- Use the following command:

```
puma{admin}# config path-failover-monitor interval {interval-value} max-try  
{max-retries}
```

Example

```
puma{admin}# config path-failover-monitor interval 5 max-try 5
```

In the preceding example, the path failover monitoring packet will be sent to the target address once in 5 seconds and will be retried 5 times before marking the interface as down.

▼ To Show the Path Failover Status

- Use the following command:

```
puma{admin}# show network
```

Sample Configuration

```
puma{admin}# config path-failover target interface 0 192.168.101.240
puma{admin}# config path-failover target interface 1 192.168.101.117
puma{admin}# config enable path-failover
```

Path-Failover-Monitor

Example show network output with path-failover enabled.

```
puma{admin}# show network

Default Gateway           : 192.168.101.1
Hostname                  : puma
DNS Primary               : Not Configured
DNS Secondary             : Not Configured
DNS Suffix                : nspg.sfbay.sun.com
Server monitor interval  : 3
Server monitor max-try   : 5
Path Failover Status      : Enabled
Path Failover Target on interface 0 : 192.168.101.240 (Path Up)
Path Failover Target on interface 1 : 192.168.101.117 (Path Up)
Path Failover monitor interval : 2
Path Failover monitor max-try : 5
```

Network Interface Table:

```
=====
If      IP Address      Mask           MAC Address      Status  Link
-----
0       192.168.101.93  255.255.255.0  00:03:ba:2c:73:9c  Up      Up
1       192.168.101.83  255.255.255.0  00:03:ba:2c:73:9d  Up      Up
=====
```

System VLAN Table:

```
=====
VLAN Type                VLAN ID      Status
-----
Management                1           Disabled
Data                       1           Disabled
=====
```

Use the `show vip` command to indicate which interface is active. With `show vip`, `IF` indicates the active interface. In this case, `IF` is `0` which indicates `0` is the active interface:

```
puma{admin}# show vip

VIP Table:
=====
VIP NAME                IP/MASK                IF
-----
1.1.1.1                 1.1.1.1/255.255.255.0  0
=====
```

The alternate interface path is the target path. A target path can be an interface on another Sun Fire B10n blade or an interface on a Sun Fire B100 blade.

Configuring Blade Failover

Before starting the failover configuration, the two interfaces on each of the two load balancers must be configured with IP addresses residing on the same subnet. The blade failover works between two load balancers. The standby blade should have the same VLAN information as the active blade configured on the Sun Fire B1600. Refer to the *Sun Fire B1600 Blade System Chassis Software Setup Guide* for the mechanism to set up the switch.

Note – You must be logged in with the access level of supervisor and the access mode of config to execute the commands in this section.

▼ To Configure the Failover Peer IP Addresses

- As `admin` in `config` mode, enter the following command, using the parameters needed:

```
puma(config){admin}# failover peer IP_address_0 IP_address_1
```

This command configures the two management IP addresses of the secondary (peer) load balancer so that all service related configurations stored on an active load balancer can be propagated to the standby load balancer. These addresses are used as the destination address for the failover monitoring.

- `IP_address_0` is the Management IP address of interface 0 of the peer load balancer
- `IP_address_1` is the Management IP address of interface 1 of the peer load balancer

The blade failover requires the deployment of the load balancers in pairs. This is the first command that needs to be executed on each load balancer.

▼ To Enable Failover Monitoring

- As `admin` in `config` mode, enter the following command:

```
puma(config){admin}# enable failover-monitor
```

This command enables the failover monitoring. Once the failover monitoring is enabled, monitoring packets are sent to the peer periodically based on the monitoring parameters.

When a failover peer IP is configured, the monitoring function is disabled by default. For the load balancers to start monitoring each other, this command must be entered on each load balancer.

▼ To Disable Failover Monitoring

- As admin in config mode, enter the following command:

```
puma(config){admin}# no enable failover-monitor
```

This command disables the failover monitoring. Once the failover monitoring is disabled, no monitoring packets are sent to the peer. As a result of this command, no packets are exchanged and the current failover state is not affected.

This command is used when no failover monitoring is desired. If this command is executed on one load balancer, the monitoring packet carries this information and in turn it causes the receiving peer to disable its failover monitoring as well. To enable the failover monitoring, the `enable failover-monitor` command must be entered from each of the two load balancers.

▼ To Start Failover

- As admin in config mode, enter the following command, using the parameters needed:

```
puma(config){admin}# failover start {local | remote}
```

This command starts the failover synchronization. The starting process includes the determination of the failover state and the synchronization of the load balancing configuration between the two load balancers.

- *local* specifies to use the local service configurations
- *remote* specifies to use the remote (peer) service configurations

This command starts the actual failover synchronization process. The local and remote modes determine which set of load balancing configurations should be used when there is no way to determine which load balancer is active. In order for the failover to function properly, the starting mode should be configured differently on each load balancer. The *local* mode implies that the load balancer contains all the necessary load balancing configurations. The *remote* mode implies that the load balancer will be configured identical to the load balancer that is started with local mode. Once the failover state is determined and the configuration is synchronized, an identical configuration number will be assigned and saved to the failover state file (`/RFA0/CONFIG/FAILOVER/config_x/failover.state`) where *x* is 1 or 2 depending on whether your load balancer is currently running `config_1` or `config_2`.

▼ To Skip the Failover Synchronization at Boot Time

At boot time, you have the option of skipping the blade failover synchronization. During boot the system prints the following message and waits for 5 seconds for you to respond.

- **Press Return when you see the following message:**

```
Press Return key to skip the failover synchronization ...
```

▼ To Stop the Failover Synchronization

- **As admin in config mode, enter the following command:**

```
puma(config){admin}# failover stop
```

This command stops the failover synchronization. Once the failover synchronization is stopped, the current failover state and status stay the same.

This command is used when no failover state synchronization is desired. This is an interactive command if it is manually invoked from the standby load balancer. It asks if the configuration files need to be erased. This assures that this load balancer will not contain any unwanted configurations when the failover is stopped.

To restart the failover functionality, the `failover start` command must be entered from the load balancer that was stopped previously.

To remove the failover configuration from the load balancer, the `remove failover` command must be entered and followed by the `commit` command.

▼ To Configure Failover Monitoring

- As admin in config mode, enter the following command, using the parameters as needed:

```
puma(config){admin}# failover-monitor interval {interval_value} max-try
{max_try_count}
```

This command configures the failover monitoring parameters. This command can be entered from the active load balancer only.

This command is used to modify the monitoring parameters. Because these two values are used to monitor the peer and detect the failure of the other load balancer, they may be adjusted based on the tolerance for an unresponsive peer. Bigger values may be used if you expect longer delay in marking the other load balancer down.

- **interval** is an optional keyword that uses the value specified in *interval_value*. *interval_value* is the time interval in seconds in which the monitoring packet is sent.
- **max-try** is an optional keyword that uses the value specified in *max_try_count*. *max_try_count* is the maximum number of tries to send the monitoring packets before taking over as active.

The default time interval is five seconds. The default maximum number of retries (**max-try**) is set to five.

▼ To Force Failover

- As admin in config mode, enter the following command:

```
puma(config){admin}# failover force-failover
```

This command forces the standby load balancer to be the active load balancer. This command is allowed only on the active load balancer.

This command is issued when you need to make the standby load balancer active. This command is not saved when the **commit** command is executed.

The failback is not implemented. If you want the original active blade which is now in standby mode to become active again, execute this command on the currently active blade.

▼ To Sync the Failover Configuration

- As admin in config mode, enter the following command:

```
puma(config){admin}# failover config-sync
```

This command manually synchronizes the load balancing configurations. This command is allowed only on an active load balancer.

Use this command when you need to resynchronize the load balancing configurations manually. This command is not saved when the `commit` command is executed.

▼ To Remove the Failover State File

- As admin in config mode, enter the following command:

```
puma(config){admin}# erase failover state-file
```

This command removes the failover state file named `failover.state` (`/RFA0/CONFIG/FAILOVER/CONFIG_x/failover.state`) where `x` is 1 or 2 depending on whether your load balancer is currently running `config_1` or `config_2`.

This command is required whenever you need to correct the failover state information and restart the failover. This command should be issued after the reboot if the failover synchronization failed because of invalid information saved in the failover state file.

▼ To Remove the Current Configuration

- As admin in config mode, enter the following command:

```
puma(config){admin}# erase failover config-lb-memory
```

This command erases the current running load balancing configuration. Use this command when you need to erase the current running load balancing configurations.

This command reinitializes all the load balancing data structures currently configured in the running memory. The purpose of this command is to start the load balancing configuration from scratch.

▼ To Remove the Failover Configuration

- As admin in config mode, enter the following command:

```
puma{config}{admin}# remove failover-config
```

This command removes the failover configuration. This command must be followed by the `commit` command to remove the failover related commands saved previously.

▼ To List the Failover Configurations

- As admin in config mode, enter the following command:

```
puma(config){admin}# show failover
Failover Information
=====
Peer IP address           : 192.50.50.142           192.50.50.143
Peer Mac address         : 00:03:ba:2c:73:a6
00:03:ba:2c:73:a7
Failover monitor interval : 5
Failover monitor max-try  : 5
Number of times state changed to Active : 0
Number of times state changed to Standby : 1

=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Standby    23             Sync      Enabled     Start       Up:Up
=====

Peer Failover Information
=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Active     23             Sync      Enabled     Start       Up:Up
=====
```

Under the If0:If1 heading in the show failover command output, Up indicates that monitoring packets are being received through the network interface. Down indicates that monitoring packets are not being received through the network interface. For example, Up:Up as shown in the previous output indicates that monitoring packets are being received over both interface 0 and interface 1.

This command lists the failover configurations and the current failover status information as follows:

- Failover Enable (If the load balancer is involved in the failover)
- Failover State (The failover role of this load balancer: active or standby)
- Configuration Number
- Failover peer information if the failover is configured.

Monitor Information:

- Monitoring Mode (Enabled or Disabled)
- Monitoring IP addresses and their status
- Monitoring Frequency
- Monitoring Retries

- Interface Status

List of Configuration Commands

TABLE 5-1 displays a list of configuration commands. If the command is marked X, this command can be initiated from both the active and standby load balancers. If it is marked Y, this command, if saved in the configuration files, will be transferred from the active load balancer to the standby load balancer when the `commit` command is entered from the active load balancer. The commands marked Y, will not be allowed on the standby blade.

When the `commit` command is executed from the active blade, several background functions will be executed in the standby blade, such as:

- Transfer the configuration files from the active to the standby
- Execute the configuration file on the standby blade
- Copy the configuration files to the configuration directory that was configured (`config_1` or `config_2`) on the standby blade.

Note – The commands listed in TABLE 5-1 do not get erased on the standby blade.

TABLE 5-1 List of Configuration Commands

Command	Active	Standby
<code>config ip interface</code>	X	X
<code>config default gateway</code>	X	X
<code>config default hostname</code>	X	X
<code>config diag level</code>	X	X
<code>config boot config</code>	X	X
<code>config boot image</code>	X	X
<code>config data vlan</code>	X	X
<code>config management vlan</code>	X	X
<code>config enable vlan management</code>	X	X
<code>config enable vlan data</code>	X	X
<code>config failover peer</code>	X	X
<code>config enable failover-monitor</code>	X	X
<code>config no enable failover-monitor</code>	X	X

TABLE 5-1 (Continued)List of Configuration Commands

Command	Active	Standby
config failover start	X	X
config failover stop	X	X
config failover config-sync	X	X
config erase failover state-file	X	X
config erase failover config-lb-memory	X	X
config failover force-failover	X	
config failover-monitor	X, Y	
config remove failover	X	X
config erase config-files	X	X
commit	X, Y	X
config build rules	X	X
config no build rules	X	X

Configuring VLAN Parameters

A virtual LAN (VLAN) is a collection of network nodes that share the same broadcast domain regardless of their physical location or connection point in the network. A VLAN serves as a logical workgroup with no physical barriers. Thus users can share information and resources as though located on the same LAN. VLANs also allow a single physical LAN to be divided into multiple logical LANs. By restricting the flow of traffic between the members of different VLANs, this allows the separation of tenants or tiers.

This chapter describes the types of VLANs available and shows how to enable and configure VLAN parameters for the Sun Fire B10n blade.

This chapter includes the following topics:

- “Available VLAN Types” on page 159
- “Enabling and Disabling VLAN Tagging” on page 160

Available VLAN Types

The Sun Fire B10n blade CLI allows for the configuration of three different VLAN types:

- **Management VLAN:**

As its name suggests, this VLAN is used for management traffic to and from the load balancing blade. Examples are Telnet traffic and signaling between the load balancer and the server. The latter includes service configuration messages and server module health monitoring. When the management VLAN is enabled, outbound management traffic is enforced for added security, and outbound management is tagged with this VLAN ID. There is one management VLAN per load balancing blade.

- Data VLAN:

This VLAN carries inbound traffic from the client network to the load balancer. The servers are expected to use this VLAN for the response traffic back to the client network. When the load balancing blade sends packets to the client network (such as during the handshake to establish a connection), it tags these packets with this VLAN ID. However, there is no inbound enforcement for the data VLAN. There is one data VLAN per load balancing blade.

- Service VLAN:

This VLAN is used for all data traffic between the load balancing blade, the server, and potentially the SSL proxy blade. Each service on the load balancing blade may have its own service VLAN configured.

Enabling and Disabling VLAN Tagging

The `enable vlan` command enables the VLAN tagging of data or management traffic from the Sun Fire B10n blade.

By default, VLAN tagging of all traffic from the blade is disabled.

When VLAN tagging is enabled, the VLAN ID used for management traffic is the one set by the `management vlan` command. The management VLAN setting also filters incoming traffic.

The VLAN ID used for tagging client traffic is the one set by the `data vlan` command.

▼ To Enable VLAN Tagging

- **As admin in config mode, enter the following command:**

```
puma(config){admin}# enable vlan all
```

The `enable vlan all` command enables VLAN tagging for both management traffic and client traffic.

- As admin in config mode, enter the following command:

```
puma(config){admin}# enable vlan management
```

The `enable vlan management` command enables VLAN tagging for management traffic only.

- As admin in config mode, enter the following command:

```
puma(config){admin}# enable vlan data
```

The `enable vlan data` command enables VLAN tagging for client traffic only.

▼ To Set Client VLAN Tagging

The `data vlan` command sets the default VLAN ID to be used on all outbound data traffic destined to and from the client.

This parameter must be set for each blade and serves as a default. This parameter is valid only if the content load balancing blade is enabled for VLAN tagging of outbound traffic by the `enable vlan` command.

The valid range for VLAN ID values is 1 to 4095.

- As admin in config mode, type the following command:

```
puma(config){admin}# data vlan vlan_id
```

Where *vlan_id* is the data VLAN tag. The default VLAN ID is 1.

▼ To Set Management VLAN Tagging

The `management vlan` command sets the management VLAN ID on the content load balancing blade. When the management VLAN is enabled, the blade processes inbound management traffic only when it is tagged with this VLAN ID. The blade also uses the VLAN ID to tag all outbound management traffic.

The valid range for VLAN ID values is 1 to 4095.

- As admin in config mode, type the following command:

```
puma(config){admin}# management vlan vlan_id
```

Where *vlan_id* is the management VLAN tag. The default management VLAN is 2.

Example

```
puma{config}{admin}# management vlan 22
```

▼ To Disable VLAN Tagging

The `no enable vlan` command disables the VLAN tagging of outbound data or management traffic from the content load balancing blade.

- As admin in config mode, type the following command:

```
puma{config}{admin}# no enable vlan <data|management|all>
```

TABLE 6-1 describes the parameters for the `no enable vlan` command:

TABLE 6-1 Parameters for the `no enable vlan` Command

Parameter	Description
<code>data</code>	Disables VLAN tagging of outbound data traffic.
<code>management</code>	Disables VLAN tagging of outbound management traffic.
<code>all</code>	Disables VLAN tagging of all outbound traffic.

By default, VLAN tagging of all outbound traffic is disabled on the content load balancing blade.

- As admin in config mode, type the following command:

```
puma(config){admin}# no enable vlan all
```

The `no enable vlan all` command disables VLAN tagging for both management and client traffic.

- As admin in config mode, type the following command:

```
puma(config){admin}# no enable vlan management
```

The `no enable vlan management` command disables VLAN tagging for management traffic only.

- As admin in config mode, type the following command:

```
puma(config){admin}# no enable vlan data
```

The `no enable vlan data` command disables VLAN tagging for client traffic only.

▼ To Enable VLAN Tagging for a Service

- As admin in config mode, enter the `enable service vlan` command:

```
puma(config){admin}# enable service vlan service_name
```

By default, VLAN tagging is disabled for a service.

Examples

```
puma(config){admin}# enable service vlan SVC1
```

▼ To Set VLAN for Service

The `service vlan` command sets the VLAN tag to be added to all the traffic of a service.

- As admin in config mode, enter the `service vlan` command:

```
puma(config){admin}# service vlan service_name vlan vlan_id
```

Where:

service_name is the name of the service.

vlan is the qualifier for the VLAN ID.

vlan_id is the VLAN ID.

The valid range for VLAN ID values is 1 to 4095. The default VLAN ID is 1.

Example

The following example uses the *vlan_id* of 25 for the service SVC1.

```
puma(config){admin}# service vlan SVC1 vlan 25
```

▼ To Disable VLAN Tagging for a Service

- As admin in config mode, enter the `no enable service vlan` command:

```
puma(config){admin}# no enable service vlan service_name
```

By default, VLAN tagging is disabled for a service.

Examples

The following example disables VLAN tagging for the service SVC1.

```
puma(config){admin}# no enable service vlan SVC1
```


▼ To Show VLANs

- As admin, enter the `show vlan` command:

```
puma{admin}# show vlan
```

You will see output similar to the following:

```
puma{admin}# show vlan
```

```
System VLAN Table:
```

```
=====
VLAN Type                VLAN ID    Status
-----
Management              22        Enabled
Data                    21        Enabled
=====
```

```
Service VLAN Table:
```

```
=====
Service Name            VLAN ID    Status
-----
SVC1                   25        Enabled
s1                     1         Disabled
=====
```


Updating the Application Software and the BSC Firmware

This chapter describes how to upgrade the software and firmware on one or more Sun Fire B10n blades. It also tells you how to set up a TFTP (Trivial File Transfer Protocol) server if you do not already have one set up on your network. The software upgrade procedures require you to use TFTP.

- “Introduction” on page 167
- “Setting up a TFTP Server” on page 168
- “Software Architecture” on page 169
- “Upgrading and Downgrading Software on the Sun Fire B10n Load Balancer” on page 170
- “Updating B10n Application Software and BSC Firmware” on page 172
- “Choosing the Boot Image” on page 176

Introduction

Note – To perform the update procedures in this chapter, you need to log into one of the System Controllers (SCs) using telnet. This is because you need to transfer the new firmware from a location on your network.

The BSC on each blade server is a management agent for the System Controller. It communicates information about the blade server it resides in to the System Controller. It also receives and processes any commands that you type into the System Controller’s command-line interface.

Follow the instructions in this chapter if you have been advised by a Sun support engineer to download new firmware onto a System Controller, blade server, or integrated switch.

Setting up a TFTP Server

The procedures for upgrading software for the Sun Fire B10n blade involve using TFTP. This means that to perform them you need to have a TFTP server available on your network.

Note – If you are using separated data and management networks, you need a TFTP server available on both networks.

To configure a Solaris system on your network to serve TFTP requests, do the following:

1. **On the system that you intend to set up as the TFTP server, log in as root.**
2. **Use a text editor to un-comment the following line in the file `/etc/inetd.conf`:**

```
tftp dgram udp6 wait root /usr/sbin/in.tftpd in.tftpd -s /tftpboot
```

3. **On the same system create a TFTP home directory by typing the following at the Solaris prompt:**

```
# mkdir /tftpboot
# chown root /tftpboot
# chmod 755 /tftpboot
```

4. **Restart `inetd` by typing:**

```
# /etc/init.d/inetd stop
# /etc/init.d/inetd start
```

5. **Verify that TFTP is working.**

To do this, use TFTP to get a file from the `/tftpboot` directory. Follow the instructions below:

- a. On the system that you are using as the TFTP server, copy any file (for example, the Solaris `/etc/release` file) to the `/tftpboot` directory.

Type the following command at the Solaris prompt:

```
# cp /etc/release /tftpboot/filename
```

Where *filename* is the name of the file you intend to make available on the TFTP server.

- b. Make the file you have just copied read-only:

```
# chmod 444 /tftpboot/filename
```

Where *filename* is the name of the file you intend to make available on the TFTP server.

Note – TFTP is not the same as FTP. It does not display the same error messages as FTP, and you cannot use the `cd` or `ls` commands (or indeed most other commands) that FTP allows you to use.

Software Architecture

The Sun Fire B10n blade provides optimized server to client response. To support this response and provide tight communications between the content load balancing blade and the B1600 blade servers a software module must be installed on each of these servers. This software module is referred to as the Blade Server Module and is loaded using the Solaris package add (`pkgadd`) process.

The content load balancing blade is based on specialized hardware including a general purpose microprocessor that runs a real time operating system. The code that runs on this processor is called the Application Software and can be updated using a TFTP process.

In addition to the general purpose processor there is a micro controller called the Blade Support Controller (BSC). The BSC is the primary interface to the Sun Fire B1600 Service Controllers (SC) and performs the Advanced Lights-out Management (ALOM) function for a given blade. These functions include powering on and off of the blades as well as monitoring functions. This is referred to as the BSC Firmware and can be updated using the “flashupdate” command which involves using TFTP.

The Sun Fire B10n software components:

- Blade server module
- B10n application software
- BSC firmware

The B10n has the capability to hold two versions of the Application Software and a diagnostic image. This allows a new image to be loaded without overwriting the active image. The blade must be rebooted to activate an image. See “Choosing the Boot Image” on page 176.

The B10n specialized hardware includes a rule based classification engine. The rules are entered through the command line interface and then compiled using a build process. See “Creating an HTTP Load Balancing Rule” on page 95.

Check the following web site to ensure you have the latest Sun Fire B10n software:

<http://www.sun.com/software/download/network.html>

See “Updating B10n Application Software and BSC Firmware” on page 172 for instructions on checking the version of the software you are currently using.

Upgrading and Downgrading Software on the Sun Fire B10n Load Balancer

To Upgrade to Version 1.1, 1.2, or 1.3.5 From Version 1.0

The components that need to be upgraded to the 1.1 version (and onwards) are:

1. The BSC firmware
2. The B10n application software consisting of the boot image and the bootrom
3. The blade server modules

Note – Up to version 1.3.5, all images are backward compatible with the configuration files, that is, a 1.3.5 image can run with a 1.2, 1.1, or 1.0 configuration file.

To Upgrade to Version 1.2 or 1.3.5 From Version 1.1

The only components that needs to be upgraded are the B10n application software boot image and the blade server modules.

To Downgrade to Version 1.0 From Version 1.1 to 1.3.5

The components that need to be downgraded to the 1.0 version are:

1. The BSC firmware
2. The B10n application software consisting of the boot image and the bootrom
3. The blade server modules

A 1.0 image is not fully compatible with a 1.1, 1.2, or 1.3.5 configuration file because of the new features supported in the newer releases. So it is recommended that a 1.0 image be run with a 1.0 configuration file.

Note – Before upgrading to the new release, always make a backup of the current configuration so it can be used if a downgrade is required.

To Downgrade to Version 1.1 From Version 1.2 to 1.3.5

The only component that needs to be downgraded is the B10n application software boot image.

A 1.1 image is fully compatible with a 1.2 configuration file, but not with a 1.3.5 configuration file because of the new features supported in the 1.3.5 release.

Updating B10n Application Software and BSC Firmware

It is important to verify that you have the latest software for the Sun Fire B10n content load balancing blade. Check the following web site for the latest software and documentation:

<http://www.sun.com/software/download/network.html>

You need to set up a TFTP boot server to update the `sc` firmware. See “Setting up a TFTP Server” on page 168.

You also need to configure the management IP address and default gateway address. See “Configuring the Networking” on page 52.

Note – If you are updating both the B10n application software and BSC firmware, be sure to update the B10n application software *first*.

The B10n software can be loaded with three different images and booted. The three images are `image 1`, `image 2`, and `diag`. These images denote software versions.

You can upgrade the software either interactively or noninteractively.

▼ To Update the B10n Application Software

With the B10n blade in the booted and running state perform the following steps:

1. **Access the Sun Fire B10n console. At the Sun Fire B1600 SC console `SC>` type:**

```
sc> console $n
```

Where `n` is the slot number of the B10n blade

2. **Login to the B10n console.**

```
Login: admin
passwd: admin
```


3. Verify the boot image and versions:

```
puma{admin}# show system
```

```
Boot Options:
```

```
=====
Config Type   Config File   Boot Image   Diag Level   Verbose Mode
-----
running       2             1 (1.3.2)    0            0
next          2             1 (1.3.2)    0            0 0=====
```

```
Image Information Table:
```

```
=====
Image  Blade  Image Type   Version   Build Date:Time  Size
-----
1      B10n   Load Balancer 1.3.2     12/05/03 : 14:53  4046868
2      B10n   Load Balancer 1.2.2     11/26/03 :12:15  4045472
diag   B10n   Diagnostics   1.1.9     10/16/03 : 15:36  2410733
=====
```

```
Flash FS /RFA0 free space = 13,033,472 bytes
```

```
puma{admin}#
```

4. Determine which image to update (image 1 or 2), and update the empty or oldest image.
5. Update the B10n application software

```
puma{admin}# update image
```

You can upgrade the software either interactively or noninteractively.

▼ To Update the Software Noninteractively

- As admin, use the following command:

```
puma{admin}# update image tftp server file image_name image location
```

The following example uses the TFTP server with the IP address of 192.50.50.201, the image name of sunfire_b10n.1.3.2, and the image at location 1.

```
puma{admin}# update image 192.50.50.201 file sunfire_b10n.1.3.2  
image 1
```

The system returns the following output, verifying the parameters entered:

```
file exist! will overwrite /RFA0/BOOTIMAGE/boot_image_1  
Start downloading sunfire_b10n.1.3.2... using TFTP  
Transferring and writing to file /RFA0/BOOTIMAGE/boot_image_1...  
please wait.  
  
puma{admin}#
```

The following example uses the TFTP server with the IP address of 192.50.50.201, the image name of sunfire_b10n.1.3.2, and the image at location diag.

```
puma{admin}# update image 192.50.50.201 file sunfire_b10n.1.3.2  
image diag
```

The system returns the following output, verifying the parameters entered:

```
file exist! will overwrite /RFA0/BOOTIMAGE/boot_image_diag  
Start downloading sunfire_b10n.1.2.2_diag... using TFTP  
Transferring and writing to file /RFA0/BOOTIMAGE/boot_image_diag  
.....  
please wait.  
  
puma{admin}#
```

▼ To Set the New Image to be the Default Image

1. Configure the desired Boot Image. At the B10n console type:

```
puma{admin}# config boot image x
```

Where *x* is the image you just updated

2. Save the updated image using the `commit` command:

```
puma{admin}# commit
commit : Are you sure to continue? [yes|no]yes
```

3. Reboot to activate the new image:

```
puma{admin} reboot
reboot: Are you sure to continue? [yes|no] yes
```

▼ To Update the BSC Firmware

1. Escape to the system controller console by typing the pound sign (#) and period (.) in rapid succession:

```
puma{admin} #.
```

Note – If the two characters are not typed in rapid succession nothing happens.

2. At the `sc` prompt, check the current version of the BSC firmware:

```
sc> showsc -v
FRU      Software Version                Software Release Date
-----
S0       v5.1.1.4-SUNW,B10n,NetBlade1    Aug 12 2003 15:31:48
```

3. At the `sc` prompt, enter the following command:

```
sc> flashupdate -s TFTP_ip-addr -f filename sn
```

Where *TFTP_ip-addr* is the TFTP server IP address, *n* is the slot number, *filename* is the file name of the image

In the following example, 192.50.50.201 is the IP address for your TFTP boot server and `/tftpboot/525-2018-05-t2.a37` is the path to the BSC firmware image on the TFTP boot server:

```
sc> flashupdate -s 192.50.50.201. -f /tftpboot/525-2018-05-t2.a37 s12
```

4. Reset the system using `resetsc` to load the new image.

Choosing the Boot Image

The boot image can be specified for the next boot and made permanent or it can be specified at boot time.

▼ To Specify and Make the Boot Image Permanent

1. As admin, configure the boot image of your choice:

```
puma{admin}# config boot image 1
```

In this example, the chosen image is 1.

2. Commit the change:

```
puma{admin}# commit
commit: Are you sure to continue? [yes|no] yes
Success!
puma{admin}#
```

▼ To Specify the Boot Image at Boot Time

During boot the system prints the following message and waits for 3 seconds:

```
Press any key to choose boot image...
```

Pressing any key prompts for the image to choose for booting.

```
Specify Image To Boot <1 | 2 | d>
```

1 - Boots image 1.

2 - Boots image 2.

d - Boots image 3.

Updating Your Server

Use the appropriate instructions for updating your Solaris or Linux servers.

▼ To Update the SPARC Solaris Server Module

1. **Download the 1.2U version of the server module software from the following site:**
<http://www.sun.com/software/download/network.html>

2. **Unzip the file:**

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-SolarisModule.zip
```

3. **Install the SPARC Solaris server module software packages:**

```
# cd path_to_unzipped_file/Solaris/sparc  
# pkgadd -d .
```

4. Restart the Solaris server module:

```
# /etc/init.d/clbctl stop  
# /etc/init.d/clbctl start
```

▼ To Update the x86 Solaris Server Module

1. Download the 1.2U version of the server module software from the following site:

<http://www.sun.com/software/download/network.html>

2. Unzip the file:

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-SolarisModule.zip
```

3. Install the x86 Solaris server module software packages:

```
# cd path_to_unzipped_file/Solaris/i386  
# pkgadd -d .
```

4. Restart the Solaris server module:

```
# /etc/init.d/clbctl stop  
# /etc/init.d/clbctl start
```

▼ To Update the Linux Server Module

1. Download the 1.2U version of the server module software from the following site:

<http://www.sun.com/software/download/network.html>

2. Unzip the file:

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-LinuxModule.zip
```

Different Linux OS subdirectories are available, such as, RHAS_2.1, SLES_8.0, ... additionally, hardware platforms such as Scimitar1P, Scimitar2P and V60_65x are available.

3. Install the Linux server module for RHAS 2.1 Update 2, for example:

```
# rpm -i sun-clb-k2_4_9_e_24-1.41-1.i386.rpm
# rpm -i sun-clb-admin-1.41-1.i386.rpm
```

4. Restart the Linux server module:

```
# /etc/init.d/clbctl stop
# /etc/init.d/clbctl start
```

▼ To Upgrade the Linux Server Module From an Existing Installation

1. Download the 1.2U version of the server module software from the following site:

<http://www.sun.com/software/download/network.html>

2. Unzip the file:

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-LinuxModule.zip
```

Different Linux OS subdirectories are available, such as, RHAS_2.1, SLES_8.0, ... additionally, hardware platforms such as Scimitar1P, Scimitar2P and V60_65x are available.

3. Upgrade the Linux server module for RHAS 2.1 Update 2, for example:

```
# rpm -U sun-clb-k2_4_9_e_24-1.41-1.i386.rpm
# rpm -U sun-clb-admin-1.41-1.i386.rpm
```

4. Restart the Linux server module:

```
# /etc/init.d/clbctl stop
# /etc/init.d/clbctl start
```


Configuring a Sun Fire B100x Linux Server Blade

This chapter describes how to configure a server blade with the Linux operating system, and contains the following sections.

- “To Configure a B100x Server Blade With Linux” on page 181
- “To Set Up a Sun Fire B100x/B200x Linux Server Blade” on page 182
- “Online Documentation” on page 186
- “Differences Between Linux Distributions.” on page 186

Sun Fire B100x Linux Server Blades

Sun Fire B100x server blades can be configured to use the Linux operating system.

▼ To Configure a B100x Server Blade With Linux

1. At the `sc` prompt, check the blade server module software:

```
sc> console Sn
```

Where `S` indicates a slot and `n` is the slot number.

2. At the Linux prompt, enter the following command:

```
# rpm -q sun-clb-admin
```

The response indicates the module version. If the module is loaded, type the following command:

```
# modinfo sun-clb | grep description
```

The following example checks the version on a Linux server blade installed in slot 10:

```
sc> console S10
Connected with input enabled on fru S10
Escape Sequence is '#' (#.)
# rpm -q sun-clb-admin
sun-clb-admin-1.35-1
# modinfo sun-clb | grep description:
description: "CLB (Sun Connection Load Balancing), v. 1.35"
```

▼ To Set Up a Sun Fire B100x/B200x Linux Server Blade

1. Download the software appropriate for your system available at:

<http://www.sun.com/software/download/network.html>

2. Unzip the Linux module zip file.

```
# /usr/bin/unzip SunFire_B10n-1_2_Update-LinuxModule.zip
```

Similar subdirectories of Linux versions are available, that is, RHAS_2.1 and SLES 8.0, and hardware platforms such as Scimitar1P, Scimitar2P and V60/65x which this software supports.

3. Install the Linux image for RHAS 2.1 Update 2.

```
# rpm -i sun-clb-k2_4_9_e_24-1.41-1.i386.rpm
# rpm -i sun-clb-admin-1.41-1.i386.rpm
```

Or, you may use the following commands to upgrade to a newer version of an already installed package.

```
# rpm -U sun-clb-k2_4_9_e_24-1.41-1.i386.rpm
# rpm -U sun-clb-admin-1.41-1.i386.rpm
```

4. Find the kernels that are installed on your system:

```
# ls /lib/modules
```

5. For each kernel you will use (for example, 2.4.20-6) find a package with a matching name and install it.

```
# rpm -i sun-clb-k2_4_9_e_3-1.36-1.i386.rpm
```

Or you may use the following command to upgrade an already installed package.

```
# rpm -U sun-clb-k2_4_9_e_3-1.36-1.i386.rpm
```

6. Connect to the configuration directory.

```
# cd /etc/sun-clb
```

7. Check your ethernet interfaces.

```
# /sbin/ifconfig -a
```

If you do not see entries for eth0 and eth1, edit the `clb.conf` file and change the `vnames0` and `vnames1` parameters so that they contain the appropriate interfaces to use for Switch 0 and Switch 1 of the chassis respectively. If multiple interfaces are

connected to a switch, list all as the value for the corresponding `vnames` variable, using a colon (:) as a separator. For example, `vnames0 = eth0` or `vnames0 = eth0:eth2`.

On Scimitar 2P systems, even interface numbers use Switch 0 and odd interface numbers use Switch 1. Also, on some Scimitar Linux releases, the names `eth0`, `eth1`, `eth2`, and `eth3` have been replaced with `snet0`, `snet1`, `snet2`, and `snet3` respectively. In this case, the default values must be changed. The `ifconfig -a` command shows which names your system uses.

8. Edit the `vip.conf` file in the `/etc/sun-clb` directory.

For each VIP address, place the address on its own line. One address not in use should have the word `ARP` after it as shown in the following example.

```
209.233.20.5
209.233.20.6
209.233.20.253 ARP
```

It is best to use a different ARP address for each server, although this is not required. The main issue is whether a router notices and generates error or warning messages.

9. Set up VLANs if needed, using the Linux `vconfig` program.

10. Set the VLAN name type to `DEV_PLUS_VID` if the same VLAN ID is used on two interfaces.

```
# vconfig set_name_type DEV_PLUS_VID
```

11. Use the `vconfig` program to add a VLAN to a particular interface.

In the following example, `vlanID 10` is on the physical interface `eth0`.

```
# vconfig add eth0 10
```

12. Configure a management IP address using the `ifconfig` command.

```
# ifconfig eth0 inet 209.233.20.1 netmask 255.255.255.0 arp broadcast
# ifconfig eth0 up
```

If you are using VLANs, configure the VLAN as in the following example.

```
# ifconfig eth0.5 inet 209.234.20.2 netmask 255.255.255.0 arp broadcast
# ifconfig eth0.5 up
```

13. Start the service with the following command.

```
# /etc/init.d/sun-clb start
```

14. Use the following command to verify that the service is running.

```
# /etc/init.d/sun-clb status
```

15. Use the following command to list the VIP addresses.

```
# /etc/init.d/sun-clb lsvip
```

The `ifconfig` command shows logical interfaces for names that start with `clb` on the loopback interface, with one entry per VIP address.

```
# ifconfig
...
lo:clb0   Link encap: Local Loopback
...
```

16. Use the following command to print statistics.

```
# /etc/init.d/sun-clb stats
```

Statistics are printed with one entry per line.

17. Use the following command to print the module configuration.

```
# /etc/init.d/sun-clb showconf
```

This command outputs the same information in the `/etc/sun-clb/clb.conf` file unless the `/etc/sun-clb/clb.conf` file was edited after the `sun-clb` service was started or unless parameters were also provided on the command line when the service was started.

18. Use the following command to start `sun-clb` automatically after booting.

```
# chkconfig --add sun-clb
```

19. Use the following command to stop `sun-clb` from automatically starting after the system boots.

```
# chkconfig --del sun-clb
```

Online Documentation

There are online manual pages for the `clb.conf`, `vip.conf`, files and the `sun-clb` script. The two configuration files in the `/etc/sun-clb` directory, `clb.conf`, and `vip.conf`, are also self-documented. Blank lines are ignored and anything after a pound character (`#`) up to and including the end of line is treated as a comment.

Use the following commands to view the online manual pages.

```
# man sun-clb
# man clb.conf
# man vip.conf
```

The `sun-clb` script has some additional features not described in the online manual page, including the ability to provide configuration changes on the command line as the module starts. Other than a change to the `vnames0` and `vnames1` variables, the defaults provided in `clb.conf` work well.

Differences Between Linux Distributions.

Check the online manual pages for `rpm`, `chkconfig`, `vconfig`, and `ifconfig` as the implementations are not identical between distributions. Some versions of the Linux kernel do not support VLANs.

On Linux Redhat systems, use the following command instead of

```
# /etc/init.d/sun-clb ...
```

```
# service sun-clb ...
```

Most Linux distributions allow Ethernet interfaces to be specified as `eth0`, `eth1`, and so on. Some distributions, however, either do not do this or do not do this by default.

Diagnostics and Troubleshooting

This appendix provides an overview of the diagnostic tools and instructions for invoking tests on the Sun Fire B10n blade. There is also a section outlining some common troubleshooting issues. This appendix contains the following sections:

- “Diagnostic Tools” on page 187
- “NPU POST and SDRAM Diagnostic Tests” on page 188
- “Image Management and Troubleshooting” on page 194
- “Setting the Diagnostic Level” on page 197

Diagnostic Tools

Diagnostics for the Sun Fire B10n blade includes NPU Power On Self Test (POST) routines, manually invoked tests, error logging routines, and debugging commands.

These diagnostics can be loosely broken down into two levels:

- Low-Level diagnostics are automatic boot time tests based on a CPU-centric view of the world. These tests are automatically done at every system bringup (boot or reset) and cannot be invoked from the CLI.
- Mid-Level diagnostics are also automatic tests and can be manually invoked through a software interface. They do a basic sanity check of all the devices on the blade.

Additional functions allow you to manually run ROM, RAM, and I/O, interface tests as well as RAM and ROM read loops.

NPU POST and SDRAM Diagnostic Tests

When you boot the Sun Fire B10n blade, you can choose from a menu to run the NPU POST, SDRAM, or LoopBack test.

- NPU POST runs automatically without user intervention.
- SDRAM allows you to run a few common tests on the memory.
- LoopBack allows you to run various LoopBack tests.

Following is an example of the diagnostics menu:

CODE EXAMPLE A-1 Diagnostics Menu

```
Copyright 2003 Sun Microsystems, Inc.
Copyright 1984-2001 Wind River Systems, Inc.

Booting SunFire B10n Blade
Bootrom Build Date: Feb 12 2004, 16:06:00

Press any key to choose configuration file option...
2
Specify Configuration To Use '1 | 2 | q' > q

Using stored configuration option

Press any key to choose boot image...
2
Specify Image To Boot '1 | 2 | d | m'> d

Booting diag image

Booting Image /RFA0/BOOTIMAGE/boot_image_diag ...2144272

Initializing RDRAM           ... Done
Initializing SDRAM ECC       ... Done
Initializing BSC Interface   ... Done
Initializing Classifier Driver ... Done

Initialization done

****Entering Diag Code ****

Image Information:
```


CODE EXAMPLE A-1 Diagnostics Menu (Continued)

Image	Blade Type	Version	Build Date	Build Time	Size

/RFA0/ diag	- Volume is OK Bl0n	1.3.2_diag	02/12/04	16:09	2502011
Puma DIAGNOSTICS		Rev: 1.3.1	01/31/2004		MAIN Menu
All	All Test Sequence				
Ram	Memory Test(SDRAM)				
LBack	Loop Back Test				
Npu	NPU Test				
Dma	NPU DMA Test				
Manufacture	Manufacturing Test				
ESC + RETURN	Stop/Previous Menu				
REBoot	Terminate PUMADIAG and Reboot				
PumaDiag ==>					

You can choose the specific tests from this menu. If you choose `r`, then the memory test is invoked and the next menu is displayed, for example:

CODE EXAMPLE A-2 Memory Test Options

Puma DIAGNOSTICS		Rev: 1.3.1	01/31/2004		MAIN Menu
All	All Test Sequence				
Ram	Memory Test(SDRAM)				
LBack	Loop Back Test				
Npu	NPU Test				
Dma	NPU DMA Test				
Manufacture	Manufacturing Test				
ESC + RETURN	Stop/Previous Menu				
REBoot	Terminate PUMADIAG and Reboot				
PumaDiag ==> Ram					
Ram					
Puma DIAGNOSTICS		Rev: 1.3.1	01/31/2004		Memory Menu
All	All Test Sequence				
Default	Default Test Sequence				
Quick	Quick Test Sequence				
March	MarchTest				
BMarch	March B Test				

CODE EXAMPLE A-2 Memory Test Options

```
FMarch Full ..... March Full Test
BFMarch Full .... MarchB Full Test

ESC + RETURN ..... Stop/Previous Menu
REBoot ..... Terminate PUMADIAG and Reboot

PumaDiag ==> March
March

****RUNNING MARCHING32 TEST from Address 0x30000000 to Address 0x30100000
Marching Test: March32 test on memory starting at 0x30000000 and ending at
0x30100000.
PASS!!. Completed Marching test

Hit [RETURN] to Enter the Memory Menu
Memory March Test 0.....PASSED
Completed PUMA Diagnostics Test
Memory March Test (p=1 f=0) PASSED

Puma DIAGNOSTICS          Rev: 1.3.1          01/31/2004          Memory Menu

All ..... All Test Sequence
Default ..... Default Test Sequence
Quick ..... Quick Test Sequence

March ..... MarchTest
BMarch ..... March B Test

FMarch Full ..... March Full Test
BFMarch Full .... MarchB Full Test

ESC + RETURN ..... Stop/Previous Menu
REBoot ..... Terminate PUMADIAG and Reboot

PumaDiag ==>
```

- Basic: This test runs the chosen memory test from the memory address location 0(zero) to 1MB that is, start address is 0 in memory and end address is 100000(hex) in memory and returns "Success" on completion or Failure and where it failed.
- Full: This test runs a full memory test on the entire memory(256MB). This is the memory that allows the test to be run.
- Specify: This allows you to run memory tests on any allowed memory region. The input values are accepted in decimal value and then translated to the corresponding HEX value.

Following is an example of the SDRAM memory test menu:

```
List of SDRAM Memory Test to run
=====

Marching Test          m
MarchB Test           b
Quit                  q
Specify the Test type :
```

Similarly other tests and Test Types can be performed. Typing q takes you to the previous menu where other tests can be performed, for example:

CODE EXAMPLE A-3 Example of Tests

```
Puma DIAGNOSTICS          Rev: 1.3.1          01/31/2004          MAIN Menu

All ..... All Test Sequence
Ram ..... Memory Test(SDRAM)
LBack ..... Loop Back Test
Npu ..... NPU Test
Dma ..... NPU DMA Test
Manufacture ..... Manufacturing Test

ESC + RETURN ..... Stop/Previous Menu
REBoot ..... Terminate PUMADIAG and Reboot
PumaDiag ==> LBack
LBack

Puma DIAGNOSTICS          Rev: 1.3.1          01/31/2004          NPU LoopBack Menu

SERdes ..... Internal Loopback
Back To Back ..... External Loopback
NPU To Focus ..... FPGA Loopback

ESC + RETURN ..... Stop/Previous Menu
REBoot ..... Terminate PUMADIAG and Reboot

PumaDiag ==>
```

CODE EXAMPLE A-3 shows the choices for the various LoopBack tests between the two Gigabit Ethernet interfaces.

- The Serdes (Internal) Loopback: This does an Internal loopback from Interface 0 to Interface 1 and back again to Interface 0. Packets of a predefined size are sent from one interface to the other and back to the originating interface and checked for correctness, errors, and so on. Then the statistics are displayed.

- Back -To -Back (External) Loopback: This requires external cabling between the two Interfaces, hence this is probably not possible on a Sun Fire B1600 chassis.
- NPU-To_Focus FPGA Loopback: This is similar to the Internal Loopback with the classifier bypassed.

```

Puma DIAGNOSTICS          Rev: 1.3.1          01/31/2004          NPU LoopBack Menu

SErdes ..... Internal Loopback
BAck To BAck ..... External Loopback
NPU To Focus ..... FPGA Loopback

ESC + RETURN ..... Stop/Previous Menu
REBoot ..... Terminate PUMADIAG and Reboot

```

Choosing SE, performs the internal loopback test. Choosing BA, performs the external loopback test. Choosing NPU, performs the FPGA loopback test.

Quitting the Diagnostics menu automatically reboots the blade. See “The Boot Process” on page 194 for how to boot the blade and choose images for booting.

After the tests complete, the board resets itself and returns to the boot prompt and chooses the default image (Image 1) to start the Sun Fire B10n blade application (the application image for doing the Load Balancing).

Additionally, you can invoke the NPU test, using the following syntax:

```
puma{admin}# dump module module_name index
```

For example:

```
puma{admin}# dump module npu 2
```

The dump module command produces output similar to the following example:

```
--- MulticastIdManager ---
Agent Base      = 40000000
Register Base   = 14030000
FBUS Errors     = 00

DBS[0] Base = 01600000
DBS[0] Size = 2

DBS[1] Base = 02600000
DBS[1] Size = 2

DBS[2] Base = 03600000
DBS[2] Size = 2

DBS[3] Base = 04600000
DBS[3] Size = 2

DBS[4] Base = 00000000
DBS[4] Size = 0

DBS[5] Base = 00000000
DBS[5] Size = 0

DBS[6] Base = 00000000
DBS[6] Size = 0

DBS[7] Base = 00000000
DBS[7] Size = 0

MID_DEALLOC_PTR      = 0000000000000000
MID_DEALLOC_DBP(0)   = 0000000000000000
MID_DEALLOC_DBP(1)   = 0000000000000000
MID_DEALLOC_DBP(2)   = 0000000000000000
MID_DEALLOC_DBP(3)   = 0000000000000000
MID_DEALLOC_DBP(4)   = 0000000000000000
MID_DEALLOC_DBP(5)   = 0000000000000000
MID_DEALLOC_DBP(6)   = 0000000000000000
MID_DEALLOC_DBP(7)   = 0000000000000000

puma{admin}#
```

Various modules will dump the Registers based on which index you choose (if present). The basic POST test is still run on all images at boot time.

Image Management and Troubleshooting

The Sun Fire B10n blade has 32 MB (16MB X 2 Devices) of flash memory. All the system images and configuration information are stored in the flash memory.

The different system images are:

- Boot Image—Bootloader image, one for each blade
- Application image—The image that has all the software to do load balancing, two images for each board and one `diag` image.

When the blade is shipped, both the images are the same. Later you can upgrade to image 1 or image 2 from the CLI and do a commit to say you want to use the latest image. When the system is rebooted the `config` file is read to pick the right image to boot. By default Image 1 is loaded by the boot loader (Boot image) unless you commit image 2 from the CLI.

Choosing the diagnostic image (Image d) allows you to troubleshoot the blade.

The Boot Process

Once the content load balancing blade is powered on in a Sun Fire B1600 blade system chassis, it goes through the boot process. The boot loader starts an `auto_boot` by counting down two seconds before it picks up the default application image (image 1) or the image committed into the configuration file.

Actually it reads the `config` file for every boot to see which image to `auto_boot`. If for some reason the boot loader cannot boot the image, it returns to the boot prompt and prompts you to specify the other known good image (if for some reason the default is corrupted). This is one reason to have two images.

Following is a boot example:

CODE EXAMPLE A-4 Boot Example

```
Copyright © 2003 Sun Microsystems, Inc.  
  
Copyright 1984-2001 Wind River Systems, Inc.  
  
Booting SunFire B10n Blade  
Bootrom Build Date: May 5 2003, 12:22:23  
  
Press any key to choose configuration file option...  
0
```

CODE EXAMPLE A-4 Boot Example (Continued)

```
Copyright © 2003 Sun Microsystems, Inc.
Press any key to choose boot image...
0
auto-booting...

Booting Image /RFA0/BOOTIMAGE/boot_image_1 ...3132576

Initializing RDRAM           ... Done
Initializing SDRAM ECC       ... Done
Initializing BSC Interface   ... Done
Initializing Classifier Driver ... Done
Initializing NPU Driver      ... Done
Initializing Lookup Pool     ... Done
Initializing Lookup Table CPU ... Done
Initializing Lookup Table NPU ... Done

Initialization done

Attaching network interface lo0... done.
Invalid device "tffs=0,00"

Using configuration directory: /RFA0/CONFIG/config_1
Classifier import failed [-712]
Application Initialization complete

Launching the Puma CLI!
Login:admin
Password:
Puma{admin}#
```

In CODE EXAMPLE A-4, note the one line of user interaction:

```
Press any key to choose boot image...
0
auto-booting...
```

You have two seconds to decide whether you want to continue with the `auto_boot` or interrupt and choose the image you want to load.

In the following example, the user chose the image to boot:

```
puma{admin}# reboot
reboot: Are you sure to continue? [yes|no] yes

Copyright 1984-2001 Wind River Systems, Inc.

Booting SunFire Content Load Balancing Blade
Build Date: Apr 3 2003, 19:39:54

Press any key to choose boot image...
1

Specify Image To Boot '1 | 2 | d'>
```

By interrupting the boot before it could `auto_boot`, you see the images you can choose from, that is, Image 1, Image 2, or the Diags Image.

In the following example, the user chose the diags image, but that image was not available. Therefore the system returned an error message. Then prompted the user to pick the right available image:

```
Press any key to choose boot image...
1

Specify Image To Boot '1 | 2 | d'> d
Error: Image <diag> boot file not found

Specify Image To Boot '1 | 2 | d'> d

Booting diag image

Booting Image /RFA0/BOOTIMAGE/boot_image_diag ...2157728

Initializing RDRAM           ... Done
Initializing SDRAM ECC       ... Done
Initializing BSC Interface    ... Done
Initializing Classifier Driver ... Done

Initialization done
```

Setting the Diagnostic Level

The default setting for the diagnostic level is 0. The default setting disables the test invocation. To run the test, you must set the `diag_level` to 1.

▼ To Set the Diagnostic Level

1. As admin, enter config mode:

```
puma{admin} # config
```

2. Set the diagnostic level to enable testing:

```
puma{config} {admin} # diag level 1
```

Note – `diag_level` is either 0 or 1 and 0 is default. You must reboot for the new setting to take effect.

Tutorial and Examples

This appendix provides a tutorial for configuring the Sun Fire B10n blade. It includes the following sections:

- “Configuring Layer 4 and Layer 7 Load Balancing” on page 202
- “Configuring a Layer 7 Service with SSL” on page 210
- “Setting Up VLAN” on page 221
- “Configuring Failover” on page 224

Exporting and Importing a Configuration

A maximum of two load balancing configurations can be stored in the B10n blade. You can export any or all of this configuration to a remote host. You can also import a configuration residing on a remote host onto the B10n blade.

Exporting a Configuration

Before exporting a configuration directory, the right directory needs to be compressed using the `tar` command. The following table shows the files or directories to be compressed and exported as needed.

TABLE B-1 Files and Directories to be Compressed and Exported

Configuration to Export	File or Directory to Compress
All	/RFA0/CONFIG
LB Config 1	/RFA0/CONFIG/config_1
LB Config 2	/RFA0/CONFIG/config_2
Failover Config	/RFA0/CONFIG/FAILOVER
Boot Options	/RFA0/CONFIG/boot_options.conf
Users	/RFA0/CONFIG/users.conf
Aliases	/RFA0/CONFIG/aliases.conf

▼ To Export the Entire Configuration

1. As `admin`, type the following command:

```
puma{admin}# cd /RFA0
```

2. Compress the file:

```
puma{admin}# tar B10nconfig.tar CONFIG
```

3. Enter the `export file` command and respond to the prompts:

```
puma{admin}# export file
The FTP server address: <ftp_server_ip>
The source directory path: type [cr] to use current
directory:
(null) source path, using current directory
The source file name: B10nconfig.tar
The destination directory path: <path_on_ftp_server>
The destination file name: B10nconfig.tar
The user name: <user_name_for_ftp_server>
The user password: <user_password_for_ftp_server>

export file succeed!
```

▼ To Import the Entire Configuration

1. As admin, type the following command:

```
puma{admin}# cd /RFA0
```

2. Enter the `import file` command and respond to the prompts:

```
puma{admin}# import file
The FTP server address: <ftp_server_ip>
The source directory path: type [cr] to use current
directory:
(null) source path, using current directory
The source file name: B10nconfig.tar
The destination directory path: <path_on_ftp_server>
The destination file name: B10nconfig.tar
The user name: <user_name_for_ftp_server>
The user password: <user_password_for_ftp_server>

import file succeed!
```

3. Uncompress the file:

```
puma{admin}# untar B10nconfig.tar
```

4. Reboot the system for the new configurations to take effect.

Caution – Do not do a `commit` before `reboot` because that would overwrite the imported configuration in flash with the one in memory.

Configuring Layer 4 and Layer 7 Load Balancing

Setting the Networking Configurations

Use the following procedures to set up the networking configurations.

▼ To Configure the IP Addresses

1. As admin in config mode, configure the IP address on interface 0:

```
puma(config){admin}# ip interface 0 192.50.50.134 mask  
255.255.255.0
```

2. As admin in config mode, configure the IP address on interface 1:

```
puma(config){admin}# ip interface 1 192.50.50.135 mask  
255.255.255.0
```

▼ To Configure the Default Gateway

- As admin in config mode, configure the default gateway:

```
puma(config){admin}# default gateway 192.50.50.200
```

▼ To Configure the DNS Server

- As admin in config mode, configure the primary DNS server:

```
puma(config){admin}# dns server 192.50.50.100 primary
```

▼ To Configure DNS Suffix

- As admin in config mode, configure the DNS suffix:

```
puma(config){admin}# dns suffix mycompany.com
```

Configuring a Basic Layer 4 Service Without Rules

▼ To Create a Layer 4 Service

1. As admin in config mode, configure a Layer 4 service:

```
puma(config){admin}# service name svcL4 vip 199.99.9.1:80:tcp  
interface 0
```

2. Verify that the service is configured:

```
puma(config){admin}# show service svcL4
```

▼ To Add Two blade servers to the Default Load Balancing Group for the Service

1. As admin in config mode, configure the default server:

```
puma(config){admin}# service lb-group default svcL4 server  
192.50.50.10:80:tcp:5:1 192.50.50.11:80:tcp:2:0
```

2. Set the standby server to active:

```
puma(config){admin}# modify service lb-group server svcL4:default  
server 192.50.50.11:80:tcp mode active
```

3. Verify that the servers are configured in the service:

```
puma(config){admin}# show service-lb-group svcL4 default
```

4. Enable this service:

```
puma(config){admin}# enable service name svcL4
```

5. Configure the VIP 199.99.9.1 on the loopback interface of the blade servers 192.50.50.10 and 192.50.50.11. Install the blade server load balancing packages and configure the blade server for load balancing. Refer to “Configuring the Blade Servers” on page 37.

Note – Now you can run traffic from clients to the Layer 4 service 199.99.9.1 and it is going to be load balanced between the 2 blade servers 192.50.50.10 and 192.50.50.11 in a round-robin (default load balancing scheme) fashion.

▼ To Add Another Server to the Default Load Balancing Group

1. As admin in config mode, add another server to the default load balancing group:

```
puma(config){admin}# service lb-group server svcL4:default server  
192.50.50.12:80:tcp:1:1
```

2. Verify that the server was added:

```
puma(config){admin}# show service-lb-group svcL4 default
```

3. Configure the VIP 199.99.9.1 on the loopback interface of the blade servers 192.50.50.12.

Note – Now you can run traffic from clients to the Layer 4 service 199.99.9.1 and it is going to be load balanced between the 3 blade servers 192.50.50.10, 192.50.50.11 and 192.50.50.12 in a round-robin (default load balancing scheme) fashion.

▼ To Remove a Server From the Default Load Balancing Group

1. As admin in config mode, remove a server from the default group:

```
puma(config){admin}# remove service lb-group server svcL4:default  
server 192.50.50.10:80:tcp
```


2. Verify that the server has been removed:

```
puma(config){admin}# show service-lb-group svcL4
```

Configuring a Basic Layer 4 Service With Rules

▼ To Configure a Layer 4 Service with Layer 4 (IP) Rule

1. As admin in config mode, add a Layer 4 rule to the system:

```
puma(config){admin}# ip-rule IP1 rule 192.50.50.200:3442 mask  
255.255.255.240:0
```

2. Verify that the rule was added:

```
puma(config){admin}# show rule
```

3. Create a Layer 4 service:

```
puma(config){admin}# service name svcL4-r vip 199.99.9.2:80:tcp  
interface 0
```

4. Verify that the service was created:

```
puma(config){admin}# show service svcL4-r
```

5. Add two blade servers to the default load balancing group for the service:

```
puma(config){admin}# service lb-group default svcL4-r server  
192.50.50.10:80:tcp:1:1 192.50.50.11:80:tcp:1:1
```

6. Configure the VIP 199.99.9.2 on the loopback interface of the blade servers 192.50.50.10 and 192.50.50.11.

7. Associate the Layer 4 service with a rule and a group of servers:

```
puma(config){admin}# service lb-group name IP-GRP service svcL4-r
server 192.50.50.13:80:tcp:5:1 192.50.50.14:80:tcp:5:1 rule IP
scheme wt-round-robin
```

8. Configure VIP 199.99.9.2 on loopback interfaces of server 192.50.50.13 and 192.50.50.14.

9. Verify that the service is configured:

```
puma(config){admin}# show service svcL4-r
```

10. Enable the service:

```
puma(config){admin}# enable service name svcL4-r
```

11. Build the rules:

```
puma(config){admin}# build rules
```

Note – Don't run traffic to this service yet, wait for the build to return with completion status.

12. Check the build status:

```
puma(config){admin}# show build status
```

When the build is completed, the completion message is printed out. Once you receive the completion message, you are ready to run traffic.

You can now run traffic from clients to the Layer 4 service 199.99.9.2 and the load is balanced between the blade servers in the following fashion depending on the source IP address and Layer 4 port.

- A request from client IP 192.50.50.195 and any port is going to be load balanced between the servers 192.50.50.13 and 192.50.50.14 in a weighted round robin manner as it matches the L4 rule IP1 in the IP-GRP load balancing group for this service.

- A request from client IP 192.50.50.180 and any port is going to be load balanced between the default servers 192.50.50.10 and 192.50.50.11 in a round robin fashion as it does not match the rule IP1 in the IP-GRP load balancing group for this service.

Configuring a Basic Layer 7 Service

▼ To Configure a Basic Layer 7 Service Without Rules

1. As admin in config mode, create a Layer 7 service:

```
puma(config){admin}# service name svcL7 vip 199.99.9.3:80:tcp
interface 0 lb-layer 7 L7-proto http
```

2. Add 2 servers to the default group

```
puma(config){admin}# service lb-group default svcL7 server
192.50.50.15:80:tcp:10:1 192.50.50.16:80:tcp:10:1 scheme wt-
round-robin
```

3. Enable the service:

```
puma(config){admin}# enable service name svcL7
```

4. Configure the VIP 199.99.9.3 on the loopback interface of the blade servers 192.50.50.15 and 192.50.50.16.

Note – You can now run traffic from client(s) to the Layer 7 service 199.99.9.3. The service is load balanced between the two blade servers 192.50.50.15 and 192.50.50.16 in the default load balancing group in a weighted round-robin fashion.

▼ To Configure a Layer 7 Service with Layer 7 Rules

1. As admin in config mode, add a Layer 7 rule to the system:

```
puma(config){admin}# http-rule HTML dynamic string *.html
```

2. Verify that the rule was added:

```
puma(config){admin}# show rule
```

3. Create a Layer 7 service:

```
puma(config){admin}# service name svcL7-r vip 199.99.9.4:80:tcp  
interface 0 lb-layer 7 L7-proto http
```

4. Add two servers to the default group:

```
puma(config){admin}# service lb-group default svcL7 server  
192.50.50.10:80:tcp:1:1 192.50.50.13:80:tcp:1:1
```

5. Associate the Layer 7 service with a rule and a group of servers:

```
puma(config){admin}# service lb-group name URL-GRP service svcL7-  
r server 192.50.50.11:80:tcp:5:1 192.50.50.12:80:tcp:5:1 rule HTML  
scheme wt-round-robin
```

6. Verify that the service was configured:

```
puma(config){admin}# show service svcL7-r
```

7. Enable the service:

```
puma(config){admin}# enable service name svcL7-r
```

8. Configure the VIP 199.99.9.4 on the loopback interface of the blade servers 192.50.50.10, 192.50.50.13, 192.50.50.11 and 192.50.50.12.

9. Build the rules:

```
puma(config){admin}# build rules
```

Note – Don't run traffic to this service yet, wait for the build to return with completion status.

10. Check the build status:

```
puma(config){admin}# show build status
```

When the build is completed, the completion message is printed out. Now we are ready to run traffic.

You can now run traffic from client(s) to the Layer 7 service 199.99.9.4. The service is load balanced between the blade servers in the following fashion depending on the URL of the HTTP request:

- The URL `http://199.99.9.4/index.html` is load balanced between the servers 192.50.50.11 and 192.50.50.12 in a weighted round robin manner as it matches the Layer 7 rule HTML in the URL-GRP load balancing group for this service.
- The URL `http://199.99.9.4/index.gif` is load balanced between the servers 192.50.50.10 and 192.50.50.13 in the default group in a round-robin manner as it does not match the Layer 7 rule HTML in the URL-GRP load balancing group for this service.

11. Add a CGI rule to the system:

```
puma(config){admin}# http-rule CGI cgi string server=*1
```

12. Associate the Layer 7 service with the CGI rule and a group of servers:

```
puma(config){admin}# service lb-group name CGI-GRP service svcL7-r  
r server 192.50.50.14:80:tcp:5:1 192.50.50.15:80:tcp:5:1 rule CGI  
scheme wt-round-robin
```

13. Configure the VIP 199.99.9.4 on the loopback interface of the blade servers 192.50.50.14 & 192.50.50.15.

14. Add a Cookie rule to the system:

```
puma(config){admin}# http-rule COOKIE cookie string server=server1
```

15. Associate the Layer 7 service with the Cookie rule and a group of servers:

```
puma(config){admin}# service lb-group name CK-GRP service svcL7-r  
server 192.50.50.16:80:tcp:5:1 192.50.50.17:80:tcp:5:1 rule  
COOKIE scheme wt-round-robin
```

16. Configure the VIP 199.99.9.4 on the loopback interface of the blade servers

Rules do not take effect until you issue a build rules command.

17. Build the rules:

```
puma(config){admin}# build rules
```

After the build returns with the completion message, you can run traffic to this service in the following manner:

- The URL `http://199.99.9.4/index.html` is load balanced between the servers 192.50.50.11 and 192.50.50.12 as it matches the HTML rule in the URL-GRP load balancing group for this service.
- The URL `http://199.99.9.4/index.gif` is load balanced between the servers 192.50.50.10 and 192.50.50.13 in the default group as it does not match any rule for this service.
- The request `http://199.99.9.4/index.txt?xxx=yyy&server=S1&aaa=b` is load balanced between the servers 192.50.50.14 and 192.50.50.15 as it matches the CGI rule in the CGI-GRP load balancing group for this service.
- The request with the following HTTP header is load balanced between the servers 192.50.50.16 and 192.50.50.17 as it matches the COOKIE rule in the CK-GRP load balancing group for this service:

```
GET /test.txt HTTP/1.0\r\nHost: 199.99.9.4\r\nUser-Agent: xfd\r\nCookie: server=server1\r\n\r\n
```

Configuring a Layer 7 Service with SSL

Note – If you plan to load balance SSL traffic using one or more SSL proxy blades with the Sun Fire B10n content load balancing blade, you **MUST** use VLANs on the Sun Fire B10n blade, the Sun Fire B100s, and the SSL proxy blades. Also note that a router is also required for the SSL proxy blade setup. Please refer to “Setting Up VLAN” on page 221 and “The Role of VLANs” on page 17.

▼ To Create SSL Devices

1. Add an SSL Device.

```
puma(config){admin}# ssl name ssl_dev1 192.50.50.100
```

2. Add a Port Pair to the SSL Device.

```
puma(config){admin}# ssl port-pair ssl_dev1 secureport 443 clearport 880
```

3. Enable the SSL Device.

```
puma(config){admin}# enable ssl name ssl_dev1
```

4. Check the SSL device configuration.

```
puma{admin}# show ssl ssl_dev1  
SSL Device Name           : ssl_dev1  
Enabled/Disabled          : Enabled  
Port pairs (Secure / Clear) : 443:880  
  
Interface Table:  
=====
```

If	Host Name/IP	Status	MAC
0	192.50.50.100	Up	00:50:c2:0b:1c:18

```
-----  
=====
```

If you have an additional SSL device that needs to be created, repeat steps 1 to 4.

5. List SSL devices.

```
puma(config){admin}# show ssl  
  
SSL Device Table:  
=====
```

SSL Device Name	Port Pair	Status	Ifs
ssl_dev1	443:880	Enabled	0

```
-----  
=====
```

If more than 1 SSL devices was created, the additional SSL devices should be listed under this command also.

▼ To Create a Load Balancing Service with SSL

1. As admin in config mode, create a Layer 7 service with SSL:

```
puma(config){admin}# service name svcL7-SSL vip 199.99.9.4:443:tcp ssl 880
interface 0 LB-layer 7 L7-proto http
```

2. Add servers to default group:

```
puma(config){admin}# service lb-group default svcL7-SSL server
192.50.50.17:443:tcp:10:1 192.50.50.18:443:tcp:10:1 scheme wt-round-robin
```

▼ To Add SSL Device to Service

1. Add the SSL device to the service:

```
puma(config){admin}# service ssl svcL7-SSL ssl ssl_dev1:active
```

2. Check the service to see if the SSL device has been added.

```
puma(config){admin}# show service svcL7-SSL

Service Information:
  Name                               : svcL7-SSL
  VIP:port:protocol                   : 199.99.9.4:443:TCP
  Load Balancing Layer/Protocol       : L7/(http)
  Status                              : Enabled
  Active                              : Yes
  VLAN ID and Status                  :
  QoS Class                           : Not Supported
  IP persistence                      : Not configured

  TCP Timestamp option                : No
  TCP SACK option                     : No
  TCP MSS option                      : 1460
  TCP Window scale factor             : 0
  TCP Starting Window                 : 1460
```



```

TCP handoff max OPEN retries      : 5
DoS Defense                       : Not Supported
Service Tracking                  : Not Configured
SSL                               : Configured
Cookie Persistence                : Not Configured

```

End Points Table:

```

=====
VIP                               Port  Protocol  If    Type
-----
199.99.9.4                       443   TCP       0     SSL
199.99.9.4                       880   TCP       0     Decrypted
=====

```

----------*-----*-----*-----*-----*

SSL Group:

```

=====
Name                             : ssl
Scheme                           : static
Total and Active servers         : 1 / 1
Decrypted Port                   : 880

```

Group Server Table:

```

=====
SSL Device Name                  If    Status
-----
ssl_dev1                        0     UP/EN/ACT
=====

```

----------*-----*-----*-----*-----*

Server Group:

```

=====
Name                             : default
Scheme                           : wt-round-robin
Total and Active servers         : 2 / 2
Rules in group                   : 0

```

Group Server Table:

```

=====
Server Name                      Port  Wt    Status      App    Conf
-----
192.50.50.17                    0     10    UP/EN/ACT   UP/EN  Done
192.50.50.18                    0     10    UP/EN/ACT   UP/EN  Done
=====

```

3. Enable the service:

```
puma(config){admin}# enable service name svcL7-SSL
```

▼ To Configure a Service for IP Persistence

1. As admin in config mode, add IP persistence to a service:

```
puma(config){admin}# service ip-persist svcL4-r mask 8 timeout 20
```

This adds IP persistence to the service where the source IP mask is specified as 8 bits and the inactivity timeout is specified as 20 minutes.

You can now run traffic from client 192.50.50.200, and depending on the rules, it is load balanced to a particular blade server, for example, server 192.50.50.13.

Any subsequent traffic from any client in the subnet 192.50.50.0 is sent back to the same server, that is, 192.50.50.13, without any load balancing. Thus the service is now “persistent” for the subnet 192.50.50.0. If no traffic is received from the 192.50.50.0 subnet for svcL4-r for 10 minutes, then the persistence expires. Now the first connection from a client in the 192.50.50.0 subnet, say 192.50.50.185, is load balanced again and it goes to server 192.50.50.11.

Note – Short timeout values do not have the expected result. Timeout values should be at least 15–20 minutes, and the granularity of timing out entries is in the order of 10 minutes.

▼ To Remove IP Persistence From a Service

● As admin in config mode, remove IP persistence from a service:

```
puma(config){admin}# no service ip-persist svcL4-r
```

▼ To Configure a Service for Tracking

- As admin in config mode, add port tracking to a service:

```
puma(config){admin}# service tracking svcL7-r track 0:443:tcp
timeout 20
```

This example adds port tracking to the service where the tracking port is specified as 443 and the inactivity timeout is specified as 20 minutes.

Now when you run traffic from client 192.50.50.200, depending on the request, it is load balanced to a particular back end server, for example, server 192.50.50.13. Any subsequent traffic from the same client destined to the VIP 199.99.9.4 and port 443 is sent back to the same server, that is, 192.50.50.13, without any load balancing. Thus the traffic to port 443 now tracks the primary service. If no traffic is received from 192.50.50.200 for svcL7-r on port 443 for 20 minutes, then the tracking expires. Now the first connection from 192.50.50.200 to the VIP 199.99.9.4 and port 80 will be load balanced to say server 192.50.50.10 and any subsequent connections from the same client, destined to either 199.99.9.4:80 or 199.99.9.4:443 go back to the same server.

▼ To Remove Port Tracking

- As admin in config mode, remove port tracking:

```
puma(config){admin}# remove service tracking svcL7-r track
0:443:tcp
```

▼ To Add an End Point Tracking to a Service

1. As admin in config mode, add an end point tracking to a service:

```
puma(config){admin}# service tracking svcL7-r track
199.99.9.5:8080:tcp timeout 30
```

This example adds end point tracking to the service where the tracking end point is specified with IP 199.99.9.5, port 8080 and protocol TCP, and the inactivity timeout is specified as 30 minutes.

2. Configure the end point VIP 199.99.9.5 on the loopback interface of all the blade servers.

Now when you run traffic from client 192.50.50.200, depending on the request, it is load balanced to a particular blade server, for example server 192.50.50.13. Any subsequent traffic from the same client destined to the vip 199.99.9.5 and port 8080 is sent back to the same server, that is, 192.50.50.13, without any load balancing. Thus the traffic to end point 199.99.9.5:8080:tcp now tracks the primary service. If no traffic is received from 192.50.50.200 for end point 199.99.9.5:8080:tcp for 30 minutes, then the tracking expires.

Now the first connection from 192.50.50.200 to the VIP 199.99.9.4 and port 80 will be load balanced to say server 192.50.50.10 and any subsequent connections from the same client, destined to either 199.99.9.4:80 or 199.99.9.5:8080 go back to the same server.

▼ To Remove End Point Tracking

- As admin in config mode, remove end point tracking:

```
puma(config){admin}# remove service tracking svcL7-r track
199.99.9.5:8080:tcp
```

Port tracking and end point tracking can also be added to a service configured with IP persistence in which case the tracking will be valid for any client from the subnet (as specified by the persistence mask) instead of one particular client.

▼ To Configure a Service for Cookie-Based Persistence

- As admin in config mode, enter the following command:

```
puma(config){admin}# service cookie-persist svcL7-r cookie PERSIST
offset 3 delim :
```

The cookie is set by the server on the client using the header
Set-cookie: PERSIST=xyzc03232axyz;\r\n

The client when it makes the next request sends this cookie in the header
Cookie: PERSIST=xyzc032320axyz;\r\n

If the cookie persistence is set as shown above, then the content load balancing blade parses and finds the string 'c032320a' in the cookie. The name is PERSIST and the offset is 3. The offset specifies how many bytes into the value to look for to find the start of the cookie. If the cookie matches the server the request is sent to server 192.50.50.10.

Currently, the delimiter value is ignored. If this configuration is saved and retrieved, the delimiter value is always set to '!'.

▼ To Remove Cookie Persistence from a Service

- As admin in config mode, enter the following command:

```
puma(config){admin}# remove service cookie-persist svcL7-r PERSIST
```

▼ To Configure a UDP Service

The following restrictions apply to a UDP services:

- UDP services can only be load balanced at Layer 4.
- Only the static load balancing scheme is supported.
- Persistence and tracking are not supported for UDP services.

Note – A UDP end point can be added for port/end-point tracking to a TCP service

1. As admin in config mode, create a Layer 4 service:

```
puma(config){admin}# service name svcupd vip 199.99.9.1:90:udp  
interface 0
```

2. Verify that the service was created:

```
puma(config){admin}# show service svcupd
```

3. Add two blade servers to the default load balancing group for the service:

```
puma(config){admin}# service lb-group default svcudp server  
192.50.50.18:90:udp:1:1 192.50.50.19:90:udp:1:1 scheme static
```

4. Enable this service:

```
puma(config){admin}# enable service name svcudp
```

You can now run UDP traffic from clients to the UDP service 199.99.9.1 on port 90 and it is load balanced between the two blade servers 192.50.50.18 and 192.50.50.19 in a static fashion. With the current implementation of the static load balancing algorithm, all UDP traffic from the same client IP will go to the same server, as long as the server is available. If the server becomes unavailable, then the new traffic goes to the next available server.

Note – It is possible to add Layer 4 IP rules to a UDP service just like any other Layer 4 service.

▼ To Configure an FTP Service

FTP service configuration on the Sun Fire B10n blade has the following restrictions:

- An FTP service can only be layer-4 load balanced.
- Assumption: The FTP server running on the blade server has the control port on 21 and data port on 20.

When configuring an FTP service ensure the following:

- The VIP end-point added has port 21 (the FTP control port) and protocol TCP.
- The load balancing layer is specified as Layer 4 and the Layer 7 protocol is specified as FTP.
- Before enabling the FTP service, add persistence to it with mask length 0 (any other mask length can be specified if such a persistence is specifically desired).

1. As admin in config mode, create an FTP service:

```
puma(config){admin}# service name svcftp vip 199.99.9.1:21:tcp  
interface 0 lb-layer 4 L7-protocol ftp
```

2. Verify that the service was created:

```
puma(config){admin}# show service svcftp
```

3. Add two blade servers to the default load balancing group for the service:

```
puma(config){admin}# service lb-group default svcftp server  
192.50.50.14:21:tcp:1:1 192.50.50.15:21:tcp:1:1 scheme wt-round-  
robin
```

4. Configure the VIP 199.99.9.1 should be configured on the loopback interface of the blade servers 192.50.50.14 and 192.50.50.15.

5. Add IP persistence with mask length 0:

```
puma(config){admin}# service ip-persist svcftp mask 0 timeout 20
```

Note – Short timeout values do not have the expected result. Timeout values should be at least 15–20 minutes, and the granularity of timing out entries is in the order of 10 minutes.

6. Enable this service:

```
puma(config){admin}# enable service name svcftp
```

You can now run FTP client(s) from client machine(s) to the FTP service on 199.99.9.1 and the FTP sessions will be load balanced between the 2 blade servers 192.50.50.14 and 192.50.50.15 in a weighted round robin fashion (for different client IPs as client persistence is set).

Note – It is possible to add L4 (IP) rule(s) to an FTP service just like any other Layer 4 service.

▼ To Add an End Point to a Service to Make it Multi-homed

1. As admin in config mode, add an end point to a service:

```
puma(config){admin}# service point svcL4 point  
199.99.9.30:80:tcp:0:0:0
```

This command adds an end point with VIP 199.99.9.30, port 80 and protocol TCP to the service svcL4. This end point inherits all properties of the service.

2. **Configure the VIP for the new end point, that is, 199.99.9.30 in this case, on the loopback interface of all the blade servers included in the service (svcL4 in this case).**
3. **If your service has an SSL device (B10p), add another service with the VIP for the new end point.**

```
CLI# create service
```

Enter a new service name when prompted and the VIP for the new end point.

Setting Up VLAN

For this example, assume the following:

- The management VLAN tag is 22 (the default for Sun Fire B1600 blade system chassis is 2)
- The client-side data VLAN tag is 21
- The service VLAN is 25
- Only the interface connected to SSC0 is used for this service.
- The management subnet is 192.50.50.0, with a mask of 255.255.255.0
- The IP address of the content load balancing blade is 192.50.50.10
- The IP address of the server on the management VLAN is 192.50.50.201
- The virtual IP address of the service is 199.99.9.1

Set Up on the Switch

The Sun Fire B10n blade and all servers for the service must be members for all three VLANs mentioned above. The ports must be set up to forward traffic on these VLANs tagged.

Refer to the *Sun Fire B1600 Blade System Chassis Software Setup Guide* for the mechanism to set up the switch.

First, the VLAN database must be edited to include the VLANs to be used in the service.

Then, the port membership rules are set using the interface switch port setup.

Note – By default, VLAN 1 is forwarded on the ports untagged. In this example, the client side VLAN is forwarded tagged.

▼ To Set Up VLAN on the Server

The server must also be a member of all VLANs mentioned above. The management VLAN is used to exchange configuration messages between the content load balancing blade and the blade server. VLAN is also used for server health monitoring.

The service VLAN is used for all data plane traffic between the content load balancing blade and the blade server.

The route from the server to the client network must use the client side VLAN. For security reasons, the server cannot bind any services to its IP address on this interface.

1. Configure the client side VLAN interface:

```
# ifconfig ce21000 plumb 10.10.10.10 netmask 255.255.255.0 up
```

2. Configure the management VLAN interface:

```
# ifconfig ce22000 plumb 192.50.50.201 netmask 255.255.255.0 up
```

3. Configure the service VLAN interface:

```
# ifconfig ce25000 plumb 110.10.10.10 netmask 255.255.255.0 up
```

4. Configure the VIP on the loopback interface:

```
# ifconfig lo0:1 plumb 199.99.9.1 netmask 255.255.255.0 up
```

The IP address on the service VLAN is never used in any traffic, however, a valid IP address must be configured.

5. Add all three interfaces to the load balanced interfaces:

```
# /opt/SUNWclb/bin/clbconfig add ce21000  
# /opt/SUNWclb/bin/clbconfig add ce22000  
# /opt/SUNWclb/bin/clbconfig add ce25000
```

6. Verify that the route to the default gateway uses interface ce21000.

```
# netstat -r
```

The `netstat -r` command displays the routing table, including the default route.

7. Set the default route:

```
# route add default 10.10.10.10 0
```

Remove any other default route shown by `netstat -r`. Note that there are other ways to set the default route to use this interface. See the *Solaris Administration Guide*.

Note – If the physical interface used were connected to SSC1, the virtual interfaces would be `ce21001`, `ce22001`, and `ce25001`, respectively.

The interface number for VLAN n on physical interface i is determined by the following formula: $1000 * n + i$

Hence, the interface name for VLAN 123 on physical interface `ce0` is `ce123000`

▼ To Set Up VLAN on a Load Balancing Blade

First, set up a service as described in (relevant chapters). Assume that the name of the service is `SVC1`.

1. As admin, in config mode, set the service VLAN:

```
puma(config){admin}# management vlan 22
```

2. Enable the VLAN service:

```
puma(config){admin}# enable vlan management
```

3. Set the client side (default) VLAN:

```
puma(config){admin}# data vlan 21
```

4. Enable the client side VLAN:

```
puma(config){admin}# enable vlan data
```

5. Set the service VLAN:

```
puma(config){admin}# service vlan SVC1 vlan 25
```

6. Enable the service VLAN:

```
puma(config){admin}# enable service vlan SVC1
```

7. Enable the service:

```
puma(config){admin}# enable service name SVC1
```

Configuring Failover

This section provides a tutorial for configuring the blade failover and describes how to verify the basic failover functionality.

Preparation of Load Balancing Blades

Before starting any of the failover commands, verify that there are no service configurations on the load balancer that require to run on the standby blade. See “List of Configuration Commands” on page 156 for service configurations that must be local to the standby blade.

On the load balancer to be configured as local, you might want to configure the service related configurations prior to enabling failover, so that all configurations will be propagated to the standby load balancer once the failover synchronization is complete. Alternately, the `commit` or the `failover config-sync` commands can be used later.

Configuring Basic Path Failover

▼ To Add a Path Failover Target Address to Interface 0

- As admin, use the following command:

```
puma{admin}# config path-failover target interface 0 192.168.101.240
```

▼ To Add a Path Failover Target Address to Interface 1

- As admin, use the following command:

```
puma{admin}# config path-failover target interface 1 192.168.101.241
```

▼ To Enable Path Failover Monitoring

- As admin, use the following command:

```
puma{admin}# config enable path-failover
```

▼ To Configure Path Failover Monitoring Parameters

- As admin, use the following command:

```
puma{admin}# config path-failover-monitor interval 5 max-try 5
```

In the preceding example, the path failover monitoring packet will be sent to the target address once in 5 seconds and will be retried 5 times before marking the interface as down.

▼ To Show the Path Failover Status

- As admin, use the following command:

```
puma{admin}# show network
```

```
Default Gateway           : Not Configured
Hostname                  : puma
DNS Primary               : Not Configured
DNS Secondary             : Not Configured
DNS Suffix                : Not Configured
Server monitor interval  : 3
Server monitor max-try   : 5
Path Failover Status     : Enabled
Path Failover Target on interface 1 : 192.168.101.240 (Path Up)
Path Failover Target on interface 0 : 192.168.101.241 (Path Up)
Path Failover monitor interval : 3
Path Failover monitor max-try : 5
```

```
Network Interface Table:
```

```
=====
If      IP Address      Mask           MAC Address     Status  Link
-----
0       192.168.101.251  255.255.255.0  00:03:ba:2c:73:a0  Up      Up
1       192.168.101.254  255.255.255.0  00:03:ba:2c:73:a1  Up      Up
=====
```

```
System VLAN Table:
```

```
=====
VLAN Type                VLAN ID        Status
-----
Management                18             Enabled
Data                       28             Enabled
=====
```

▼ To Disable Path Failover Monitoring

- As admin, use the following command:

```
puma{admin}# config no enable path-failover
```

▼ To Remove a Path Failover Target Address on Interface 0

- As admin, use the following command:

```
puma{admin}# config remove path-failover interface 0
```

▼ To Remove a Path Failover Target Address on Interface 1

- As admin, use the following command:

```
puma{admin}# config remove path-failover interface 1
```

Configuring Basic Blade Failover

Minimum Required Commands

The following set of minimum failover commands initially starts the failover. Each command must be entered from each load balancer in sequence in order for the failover monitoring and synchronization to work. The `show failover` command may be used to list the information about the current failover configuration, status, and state.

1. `config failover peer {IP_address_0} {IP_address_1}`
2. `config enable failover-monitor`
3. `config failover start {local | remote}`

Use the `failover-monitor` command to modify the default monitoring parameters.

▼ To Set Up the Peer IP Addresses on Both Blades

- Use the `failover peer` command:

```
puma-140(config){admin}# failover peer 192.50.50.142 192.50.50.143
```

```
puma-142(config){admin}# failover peer 192.50.50.140 192.50.50.141
```

▼ To Enable Failover Monitoring on Both Blades

- Use the `enable failover-monitor` command:

```
puma-140(config){admin}# enable failover-monitor
```

```
puma-142(config){admin}# enable failover-monitor
```

▼ To Start Failover Synchronization on Both Blades

- Use the `failover start` command:

```
puma-140(config){admin}# failover start local
```

```
puma-142(config){admin}# failover start remote
```

On each load balancer, `Synchronizing Failover State ...` will be displayed. Once the failover state is determined, `Failover state is set to standby` or `Failover state is set to active` will be printed.

Note – Enter the `commit` command if you want to save the failover configuration.

Note – The above three commands, `failover peer`, `enable failover-monitor` and `failover start local` (`failover start remote`) must be entered in sequence to ensure that the failover state synchronization will be started properly.

▼ To Show the Configured Failover Information on Both Sides

- Use the `show failover` command on the active peer:

```
puma-140(config){admin}# show failover
Failover Information
=====
Peer IP address           : 192.50.50.142 192.50.50.143
Peer Mac address         : 00:03:ba:2c:73:6e 00:03:ba:2c:73:6f
Failover monitor interval : 5
Failover monitor max-try  : 5
Number of times state changed to Active : 1
Number of times state changed to Standby : 0

=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Active     1             Sync      Enabled     Start       Up:Up
=====

Peer Failover Information
=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Standby    1             Sync      Enabled     Start       Up:Up
=====

puma1{admin}#
```

- Use the `show failover` command on the standby peer:

```
puma-142(config){admin}# show failover

Failover Information
=====
Peer IP address           : 192.50.50.140 192.50.50.141
Peer Mac address         : 00:03:ba:2c:73:9c 00:03:ba:2c:73:9d
Failover monitor interval : 5
Failover monitor max-try  : 5
Number of times state changed to Active : 0
Number of times state changed to Standby : 1

=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Standby    1              Sync      Enabled     Start       Up:Up
=====

Peer Failover Information
=====
State      Config Number Config Sync  Monitoring  Start/Stop  If0:If1
-----
Active     1              Sync      Enabled     Start       Up:Up
=====

puma-142(config){admin}#
```

Note – Config number -1, means failover state-file does not exist, or invalid state. Config number 0, means no configuration has been sync-ed up or initial state. Config number 1 (or higher), means both peers are sync-ed up.

▼ To Disable Failover Monitoring on Either Blade

- Use the `no enable failover-monitor` command:

```
puma-140(config){admin}# no enable failover-monitor
```

Note – To enable monitoring, `enable failover-monitor` must be entered on each blade.

▼ To Stop Failover Synchronization on Both Blades

- Use the `failover stop` command:

```
puma-140(config){admin}# failover stop
```

```
puma-142(config){admin}# failover stop
```

▼ To Set Up the Failover Monitoring Parameters on the Active Blade

- Use the `failover-monitor` command:

```
puma-140(config){admin}# failover-monitor interval 10 max-try 10
```

▼ To Force the Standby to Active on the Active Blade

- Use the `failover force-failover` command:

```
puma-140(config){admin}# failover force-failover
```

▼ To Sync Up the Configurations on the Active Blade

- Use the `failover config-sync` command:

```
puma-140(config){admin}# failover config-sync
```

▼ To Remove the Failover State File on Either Blade

- Use the `erase failover state-file` command:

```
puma-140(config){admin}# erase failover state-file
```

This command can be used to remove the failover state file if the two load balancer blades configured for failover goes out of sync for some reason. Erasing the state-file will initiate a synchronization process between the two blades. This command will not be allowed when both the load balancers are in sync. Also removing the failover configuration will remove this file automatically.

When removing the failover state-file, the Config number becomes -1 in the `show failover` command output. Config number -1 indicates that the failover state-file does not exist. The following is an example:

```
puma-140{admin}# config erase failover state-file
erase : Are you sure to continue? [yes|no] yes

Failover State will be resynchronized. Do you want to continue? [yes/no] yes
Failover state file does not exist.
Success!
```

▼ To Remove the Running Load Balancing Configuration on Either Blade

- Use the `erase failover config-lb-mem` command:

```
puma-140(config){admin}# erase failover config-lb-mem
```

▼ To Remove the Failover Configuration on Either Blade

- Use the `remove failover-config` command:

```
puma-140(config){admin}# remove failover-config
```

Failover Synchronization and the `commit` Command

Use the `show failover` command on both load balancers to check the failover state and configuration file synchronization. If everything is configured properly, one blade should have a state of active while the other blade is set to standby. Both load balancers should have identical load balancing configurations.

You may run commands such as `show service` and `show rule` to confirm that both blades are configured identically.

If the `commit` command is executed on the active load balancer, the synchronization between these two active and standby load balancers will be done automatically.

If any of load balancing service related commands are issued on the active load balancer without being followed by the `commit` command, the standby load balancer will not be updated with any of those configurations.

Displaying Failover Module Information

Use the `dump module failover` command to dump the information regarding the failover module to the screen.

Use the `dump module task` command to dump information regarding the failover synchronization task and the failover monitoring task.

▼ To Dump Monitoring Information

- Use the `dump module failover 1` command:

```
puma-140{admin}# dump module failover 1
```

▼ To Dump the Ramdisk Directory

- Use the `dump module failover 2` command:

```
puma-140{admin}# dump module failover 2
```

▼ To Dump Failover Information

- Use the `dump module failover 3` command:

```
puma-140{admin}# dump module failover 3
```

▼ To Dump the Failover Synchronization Task

- Use the `dump module task 9` command:

```
puma-140{admin}# dump module task 9
```

▼ To Dump the Failover Monitoring Task

- Use the `dump module task 10` command:

```
puma-140{admin}# dump module task 10
```


Enterprise VLANs

This appendix provides an example for configuring the Sun Fire B10n blade with VLANs in the enterprise.

Enterprise Example Deploying VLANs

The following is a review of the VLAN capabilities of the Sun Fire Blade chassis and its components including the Content load balancing and SSL proxy blades. Detailed below is an example using VLANs in a typical enterprise configuration including the command lines to configure the individual components of the chassis. It is expected that some knowledge of the B1600 chassis and its components has already been attained. We have chosen the enterprise model as the example as it represents a common configuration as well as providing concepts that can either be expanded to or simplified as required. As security is best deployed throughout a data center we will focus on how the B1600 and its components may best fit into this larger framework.

This section does not include detail on the SSL or TLS security protocols. The SSL Proxy Blade offloads servers from compute intensive cryptographic processing and provides a secure centralized location for key and certificate administration and configuration. The decrypted traffic can be protected using VLANs and this section describes a suggested configuration for doing this.

The B1600 chassis employs both logical and physical separation features to enable protection and isolation. The first level is the separation of management and data traffic. Management access can either be by direct ethernet or serial connection to the System Service Controller (SSC) ports or inband over the ethernet switch fabric. The SSC uses a Layer 3 filter between the internal switch fabric and the system controller. An SSH agent is employed on the System Controller (SC) to provide authentication and cryptographic protection. The SSC is connected to each chassis blade by redundant point to point serial connections providing an out of band path to each

blade. Individual blade consoles can be accessed inband over the switch fabric as well. IEEE 802.1Q VLANs are available to logically separate the management traffic from the data traffic as well to segregate data traffic from different users.

When using the Sun Fire B10n content loadbalancing blade traffic can be divided into loadblancing groups and packets are forwarded based on services. Services can then be assigned a unique VLAN ID and the loadbalancer will tag traffic to the servers. If https is one of the types of traffic within a service the Sun Fire B10p SSL Proxy will return the decrypted ingress traffic to the B10n for the loadbalancing decision and will add the service VLAN tag before sending them to the server. When data leaves the servers headed for the client the packets are tagged with the data VLAN ID thus isolating this traffic from the service and management traffic. The following figure shows the configuration for our example and highlights the concepts described above.

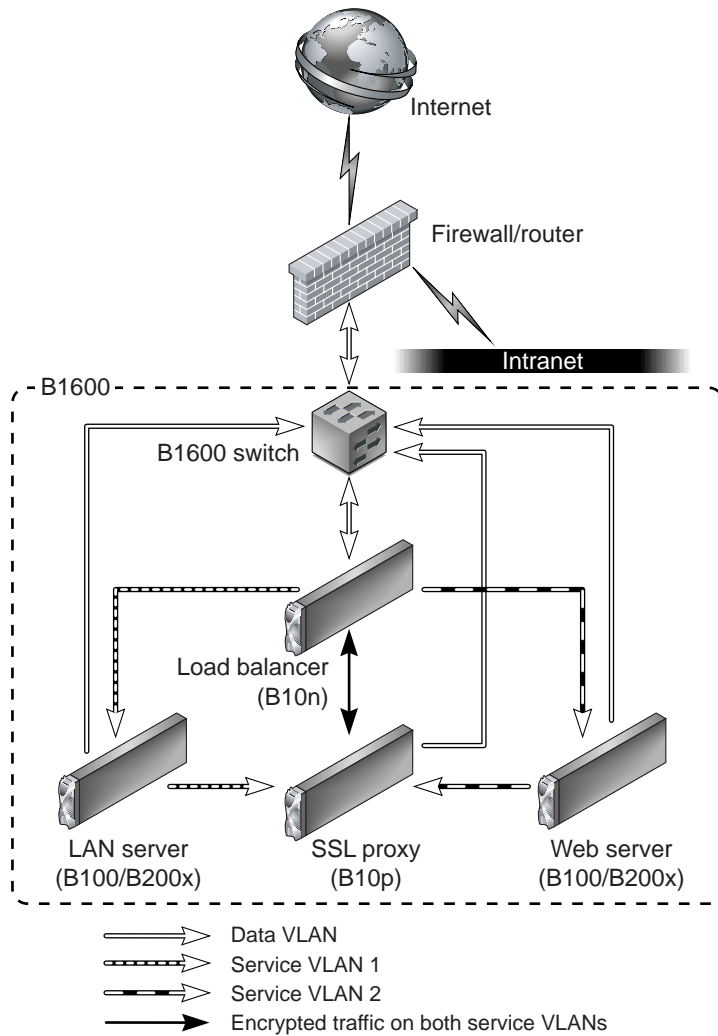


FIGURE C-1 Enterprise Security Model

Once traffic leaves the B1600 switch toward the external network the data VLAN tag can be either stripped or left and extended to the next hop router or firewall. Traffic ingress to the B1600 can be tagged by the router or firewall or can be tagged by the SSC.

While the diagram above is useful in understanding the VLAN membership concepts and logical packet flow it does not show the actual physical port connections. For example the Loadbalancer has one physical interface per switch. Therefore the switch interface must be configured to allow all three VLANs in our example. This is true for the SSL Proxy as well. The Server (or loadbalance group)

interfaces must be configured with the management, data and its service VLAN IDs. In this way the this server is isolated from the other service traffic through the switches. It is important to know that the links between the blades and the switch are point to point therefore, the switch vlan filtering can prevent other vlan traffic from being forwarded to a particular link.

User Access Control

User Access levels are provided to restrict features from different administration users. The B1600 Service controller allows multiple username and password accounts. Users are assigned up to four user access privileges types u, a, c, r which allow setting user accounts (u), administration of system controller configuration (a), console access to blades and switches (c) and the ability to power on/off blades and reset them (r).

Once you have console access, the server blades provide Unix access level options based on the type of OS you are running. In general root access is required to configure the VLAN parameters.

For the B10n Content Loadbalancing Blade there are two levels, Level 1 is limited to read access to system information and status, Level 2 is full access.

The B10p SSL Proxy blade employs three levels, Service Officer (SO) which provides full access, Administrator which allows access to network configuration but does not allow access to keys or certificates and third is User which allows read access to some system information.

Definition of VLAN Terms

1. Ingress (RX) – Packets/Frames entering the device or network being described. In data/center or server centric discussions sometimes referred to as client side or client traffic. That is traffic transmitted by the client and received by the data center.
2. Egress (TX) – Packets/Frames leaving the local device or network. In data/center or server centric discussion it is traffic transmitted by the device or network and received by the client.
3. Native – Provided to allow network control traffic (like IGMP) to travel through VLAN enabled fabrics. From a switch perspective untagged ingress packets are tagged with a specified value and egress packets with a specified ID are sent untagged.
4. Allowed – If ingress filtering is enabled the switch will allow only tagged packets specified in the list for that port. If tagged is specified it will leave tags on egress. If untagged is specified it will strip tags on egress.

5. GVRP – VLAN Registration Protocol defines the way for switches to exchange information in order to automatically register members on interfaces across a network.

System Components and VLAN Attributes

B1600 Switch

IEEE 802.1Q – Up to 256 VLANs - GVRP and manual configuration

- Management Port - NETGMT [Default VLAN ID set to 2, Ingress filtering, Egress tagging]
- Native VLAN support [Default PVID set to 1 on NETP0-8 & SNP0-15 ports]
- Port Mode [Default Hybrid or can be set to Trunk]
- Ingress filtering (VLAN tag enforcement) [Default disabled]
- Ingress Tagging [packets are always tagged within the switch fabric]
- Egress Tagging [switchport allowed] (untagged - strip specified ID, tagged - leave specified ID)

Note – The switch supports Port and Hybrid VLAN configurations. All configurations in this example use Hybrid mode where tagged and untagged VLAN IDs are specified.

B10n Content Load Balancer

(Default is all VLANs disabled)

- Management VLAN (Default VLAN id set to 2, Ingress Tag enforcement, Egress tagging, One Management VLAN per B10n)
- Data VLAN (Default VLAN id is 1, Egress tagging for client bound packets, One Data VLAN per B10n)
- Service VLAN (One VLAN per Service, Egress replacement with Service VLAN ID)

B10p SSL Proxy

Role is to decrypt incoming SSL traffic and encrypt outgoing SSL.

(Default is VLANs enabled)

- Management VLAN (Default VLAN id set to 2, Ingress Tag enforcement, Egress tagging, One Management VLAN per B10p)
- Data VLAN (Default VLAN ID is 1, Ingress Tag enforcement, Egress replacement with Service VLAN ID)
- Service VLAN (Default VLAN ID is 1, Ingress Tag enforcement, Egress replacement with Data VLAN ID)

Server Blade

- Management VLAN [Ingress filtering (Tag enforcement), Egress tagging] (must be configured for B10n control and monitoring messages)
- Data VLAN (Ingress Tag enforcement, Egress Tagging)
- Service VLAN (Ingress Tag enforcement, Egress Data VLAN Tagging for client bound, Egress Service tagging for SSL clear text to Proxy.)

Enterprise Example

FIGURE C-2 provides an example VLAN configuration of four VLANs.

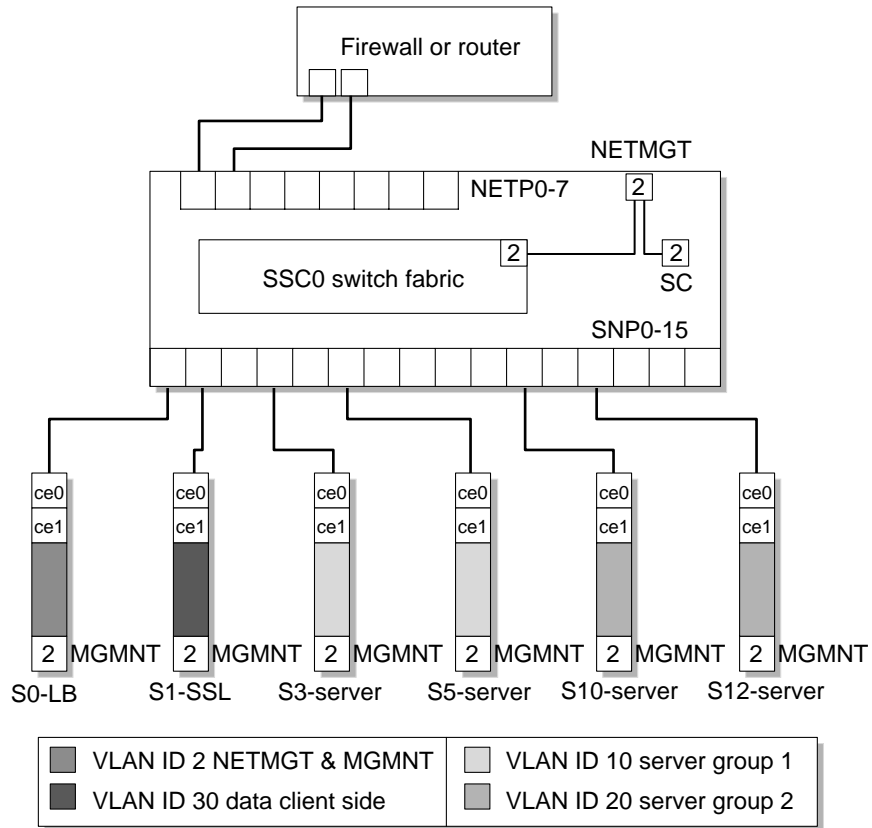


FIGURE C-2 Enterprise Example

For the enterprise example figure xxx shows a physical view including network links, Loadbalancer blade (LB), SSL Proxy blade (SSL), and two server groups each with a unique service (SVC1 & SVC2). One for Internet access and the other for Intranet traffic. VLANs IDs are only shown for the end points of the management network as this is common to all blades and switch ports. The load balancer and SSL Interfaces must be a member of all four VLANs. The Server Interfaces are members of management, data and a service VLAN. Below is a example of commands to configure the B1600 and its components based on the figure xxxx. The exception being we will not extend VLANs to the external firewall but configure the switch to untag the egress Data VLAN packets and Tag the Ingress Data VLAN packets.

The following four VLANs IDs will be used.

1. Management VLAN ID 2
2. Data (or client side) VLAN ID 30
3. Service VLAN ID 10

4. Service VLAN ID 20

SSC VLAN Configuration

To configure the system the first step will be configuring the VLANs for the SSC. The external network or “client side” uplink ports (NETP_x), the internal (SNP_x) ports and the management ports configurations will be shown. The approach we will use for the uplink (NETP_x) ports is enable ingress filtering allowing only untagged VLAN IDs. The SSCs employ the concept of “native” VLANs which allow incoming untagged packets to be tagged with the VPID or native tag and untag egress packets that contain the VPID. The approach we will use for the internal (SNP_x) ports is to allow both service VLANs as well as the management and data VLANs for the Load balancer and SSL Proxy blades. Server blade ports also allow management and data VLAN, but only one service VLAN is allowed. If all elements are in the same shelf, this contains the service VLANs (and thus cleartext and other service traffic) within the B1600. If external servers (or multiple shelves) are used, these VLANs can be extended by adding these VLAN IDs on those uplink ports connected to the additional servers or shelves. GVRP will be used in this example.

The interfaces to configured on the switch are:

- Management VLAN
- Uplink Network Ports
- Internal Switch Ports

For details on SSC switch configuration see the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*.

1. Go to the SSC console from the SC prompt.

```
sc>console sscn/swt
```

Where *n* is either 0 or 1 for systems with redundant switches.

2. Type your user name and password.

User access needs to at least have [a] administration privileges set.

```
Console#config
```

3. Configure the Management port (this step can be skipped if default vlan 2 has not been changed)

```
Console(config)#interface ethernet NETMGT
Console(config)#switchport allowed vlan add 2
Console(config)#switchport native 2
Console(config)#switchport allowed vlan remove vlan id
```

Where *vlan id* is the previously configured value.

4. Setup the VLAN database.

```
Console(config)#vlan database
Console(config-vlan)#vlan 30 name external media ethernet
Console(config-vlan)#vlan 10 name loadgrp1 media ethernet
Console(config-vlan)#vlan 20 name loadgrp2 media ethernet
Console(config-vlan)#exit
```

5. Configure the Uplink ports.

```
Console(config)#interface ethernet netp0
Console(config-if)#no switchport gvrp
```

Note – By default the `gvrp` is disabled for all ports. Anytime you change the port configuration you must include the `no switchport gvrp` command to restore `gvrp`.

6. Configure this uplink port to untag egress packets for these VLANs.

```
Console(config-if)#switchport allowed vlan add 10,20,30 untagged
```

7. Configure this uplink port to tag ingress untagged packets with this PVID.

```
Console(config-if)#switchport native vlan 30
```

8. Remove the default data VLAN.

```
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat this step for all uplink ports you want to include in the shared Data VLAN (30).

9. Configure the internal port for the Loadbalancer and SSL proxy blades.

```
Console#config
Console(config)#interface ethernet snp0
Console(config-if)#switchport gvrp
Console(config-if)#switchport allowed vlan add 10 tagged
Console(config-if)#switchport allowed vlan add 20 tagged
Console(config-if)#switchport allowed vlan add 30 tagged
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat the above instructions (10) for the SSL Proxy changing snp0 to snp1.

10. Configure the internal ports for the servers.

```
Console#config
Console(config)#interface ethernet snp3
Console(config-if)#switchport gvrp
Console(config-if)#switchport allowed vlan add 2 tagged
Console(config-if)#switchport allowed vlan add 10 tagged
Console(config-if)#switchport allowed vlan add 30 tagged
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat this sequence for the second server changing snp3 to snp5.

Repeat this sequence for the second service group changing the first allowed VLAN from 10 to 20.

The Loadbalancer VLAN Configuration

Service VLANs The B10n load balancers must be a member of all four VLANs. The management VLAN must be the same for all blades to be loadbalanced as monitoring messages go between them and the B10n. The two services will be named SVC1 and SVC2 for the two loadbalancing groups.

The steps to configure VLANs on the B10n are as follows:

1. Go to the B10n console from the SC prompt.

```
sc>console sn
```

Where *n* is the slot number of the load balancing blade.

2. Type your user name and password.

User access privileges must be Level 2.

3. Enter the B10n configuration mode.

```
puma{admin}# config
```

4. Set the management vlan and enable.

```
puma(config){admin}# management vlan 2  
puma(config){admin}# enable vlan management
```

5. Set the data vlan and enable.

```
puma(config){admin}# data vlan 30  
puma(config){admin}# enable vlan data
```

6. Set the service VLAN for SVC1 and enable.

Note – The service SVC1 and SVC2 must be configured prior to these commands. See the Sun Fire B10n Content Loadbalancing Administration Guide for details.

```
puma(config){admin}# service vlan SVC1 vlan 10  
puma(config){admin}# enable service vlan SVC1  
puma(config){admin}# enable service name SVC1
```

7. Set the service vlan for SVC2 and enable.

```
puma(config){admin}# service vlan SVC2 vlan 20
puma(config){admin}# enable service vlan SVC2
puma(config){admin}# enable service name SVC2
puma(config){admin}# exit
```

8. Verify the configuration.

```
puma{admin}# show vlan
System VLAN Table:
=====
VLAN Type VLAN ID      Status
-----
Management                2      Enabled
Data    30      Enabled
=====
Service VLAN Table:
=====
Service Name VLAN ID      Status
-----
SVC1                10      Enabled
SVC2                20      Enabled
=====
```

The SSL Proxy Blade Configuration

The SSL Proxy blade must be configured for all four vlans. The default is VLANs are enabled. If the blade has been configured and you are just adding VLANs you will begin by enabling VLANs.

1. Access the SSL proxy blade console from the SC command line interface.

```
sc>console sscn/swt
```

Where *n* is the slot number for the SSL proxy blade.

2. Type your user name and password.

User access needs to be Security Officer (SO) or Administrator (admin).

3. Enable VLANs.

```
CLI# set vlan filter enable
```

Note – Non-zero VLANs tags cannot be set before vlans are enabled

4. Configure the management VLAN.

```
CLI# set vlan management 1 2
```

5. Configure the data (client side) VLAN.

```
CLI# set vlan client 30
```

6. Configure the SVC1 Service VLAN.

```
CLI# set vlan inband 1 10
```

7. Configure the SVC1 Service VLAN.

```
CLI# set vlan inband 2 20
```

8. Verify the configuration.

```
CLI# show vlan client
      client vlan:30

CLI# show vlan manage
      port 1:
        vlan:2
      port 2:
        vlan:2

CLI# show vlan inband
      port 1:
        vlan:10
      port 2:
        vlan:20

CLI# show vlan filter
      vlan filter: enabled
```

The Solaris Server VLAN configuration

The command example here is for Solaris if Linux is being used the syntax will vary slightly but the concepts are the same.

The server must also be a member of three VLANs, the management, the data (client side) and service VLAN. The management VLAN is used to exchange configuration and monitoring messages between the content load balancing blade and the blade server. The service VLAN is used for all data traffic between the content load balancing blade and the blade server.

The route from the server to the client network must use the data (client side) VLAN. For security reasons, the server cannot bind any services to its IP address on this interface.

- The management subnet is 192.50.50.0, with a mask of 255.255.255.0
- The IP address of the content load balancing blade is 192.50.50.10
- The IP address of the server on the management VLAN is 192.50.50.201
- The virtual IP address of the service is 199.99.9.1
- The default route is set to 10.10.10.10

1. Go to the server blade console from the SC prompt.

```
sc>console sn
```

Where *n* is the slot number of the server blade.

2. Type your user name and password.

User access privileges must be superuser.

3. Configure the data VLAN interface.

```
# ifconfig ce30000 plumb 10.10.10.10 netmask 255.255.255.0 up
```

4. Configure the management VLAN interface:

```
# ifconfig ce2000 plumb 192.50.50.201 netmask 255.255.255.0 up
```

5. Configure the SVC1 service VLAN interface:

```
# ifconfig ce10000 plumb 150.10.10.10 netmask 255.255.255.0 up
```

6. Configure the VIP on the loopback interface:

```
# ifconfig lo0:1 plumb 199.99.9.1 netmask 255.255.255.0 up
```

The IP address on the service VLAN is never used in any traffic, however, a valid IP address must be configured.

7. Add all three interfaces to the load balanced interfaces:

```
# /opt/SUNWclb/bin/clbconfig add ce2000  
# /opt/SUNWclb/bin/clbconfig add ce10000  
# /opt/SUNWclb/bin/clbconfig add ce30000
```

8. Verify that the route to the default gateway uses interface ce30000.

```
# netstat -r
```

The `netstat -r` command displays the routing table, including the default route.

9. Set the default route:

```
# route add default 10.10.10.10 0
```

Remove any other default route shown by `netstat -r`. Note that there are other ways to set the default route to use this interface. See the *Solaris Administration Guide*.

Note – If the physical interface used were connected to `ssc1`, the virtual interfaces would be `ce2001`, `ce10001`, and `ce30001`, respectively.

The interface number for VLAN n on physical interface i is determined by the following formula: $1000 * n + i$

Hence, the interface name for VLAN 123 on physical interface `ce0` is `ce123000`

These steps need to be repeated for the other servers in the loadbalancing group. For the second loadbalancing group the VIP must be different and the service vlan changes from `ce10000` to `ce20000`.

Alphabetical Command Reference

This appendix provides an alphabetical listing of all the Sun Fire B10n blade commands.

A

alias Create an alias.

B

boot Boot the system.

boot config Set the boot configuration.

boot image Configure the system image to use for the next time the content load balancing blade goes through a system reboot.

broadcast Used to send a message to all the users logged into the Sun Fire B10n blade.

build rules Activate the rules just configured.

C

cat Output a file to the screen.

- chkdsk** Verify that the Flash File System is in good condition. The `check` option determines the condition of the Flash File System. The `repair` option fixes problems found in checking process.
- clear** Clear the screen.
- config** Enter configuration mode.
- commit config** Activate the configuration just entered.

D

- debug module** Configure the debug level for a specified module in the system.
- default gateway** Configure a default gateway on the load balancing blade.
- default hostname** Delete service.
- default qos level** Set the default QoS level.
- default tcp-dos-params** Set the default TCP-dos parameters.
- default tcp-handoff-params** Set the default TCP-handoff parameters.
- default tcp-params** Set the default TCP parameters.
- diag level** Configure the diagnostics level and the level of verbosity of the diagnostics.
- dns server** Configure a DNS for the load balancing blade. When supplied with a hostname, this DNS will resolve it and obtain the corresponding IP address.
- dns suffix** Set the suffix to be added to the hostnames before resolution, for example `.com`.
- dump config** Display the existing configuration.
- dump memory** Dumps the system memory to the screen.
- dump module** Dump the information about a specific module to the screen.

E

echo	Echo text typed in.
enable failover-monitor	Enable the failover monitoring.
enable path-failover	Enable path failover monitoring.
enable server	Enable the server.
enable service	Enable the configured service.
enable service app-monitor	Enable the service application monitor.
enable service tcp-dos	Enable the TCP dos service.
enable service vlan	Enable the VLAN service.
enable vlan	Enable the VLAN setting.
erase failover config-lb-memory	Erase the current running load balancing configuration. Use this command when you need to erase the current running load balancing configurations.
erase failover state-file	Remove the failover state file named <code>failover.state</code> (<code>/RFA0/CONFIG/FAILOVER/CONFIG_x/failover.state</code>) where <code>x</code> is 1 or 2 depending on whether your load balancer is currently running <code>config_1</code> or <code>config_2</code> .
exit	Exit this mode.
export config	Export a configuration.

F

failover config-sync	Manually synchronize the load balancing configurations. This command is allowed only on an active load balancer.
failover force-failover	Force the standby load balancer to be the active load balancer. This command is allowed only on the active load balancer.
failover-monitor	Configure the failover monitoring parameters. This command can be entered from the active load balancer only.

- failover peer** Configure the two management IP addresses of the secondary (peer) load balancer so that all service related configurations stored on an active load balancer can be propagated to the standby load balancer. These addresses are used as the destination address for the failover monitoring.
- failover start** Start the failover synchronization.
- failover stop** Stop the failover synchronization.
-

H

- help** Show command help.
- history** Show command history.
- http-rule** Set an HTTP rule.
-

I

- import** Import current configuration.
- ip** Configure the real IP address for the content load balancing blade to be used for management and control, for example opening a Telnet session.
- ip-rule** Set an IP rule.
-

L

- login** Log in to a user session.
- logout** Log out from a user session.
- ls** List the content of the current directory.

M

management vlan	Create the management VLAN.
modify service lb-group scheme	Modifies the load balancing scheme of a load balancing group.
modify service lb-group server	Make changes to the service for a load balancing group server.
modify service lb-group weight	Modifies the server's weight in a specified load balancing group.
modify service ssl mode	Make changes to the service for the SSL mode.
mkdir	Create a new directory.
mv	Move a file to another directory.

N

no build rules	Remove the rules just configured.
no default gateway	Remove the default gateway.
no dns server	Remove the DNS server.
no dns suffix	Unset the DNS suffix to be used by the DNS resolver.
no enable failover-monitor	Disables the failover monitoring.
no enable path-failover	Disable path failover monitoring.
no enable server	Remove the server.
no enable service app-monitor	Disables application monitoring for the service.
no enable service tcp-dos	Remove the TCP DoS service.
no enable service vlan	Remove the VLAN service.

- no enable vlan** Remove the VLAN.
 - no ip** Remove the IP address.
 - no service cookie-persist** Remove the cookie-based persistence service.
 - no service ip-persist** Remove the IP persistence service.
-

P

- password** Change the current password.
 - path-failover-monitor** Configure path failover monitoring parameters.
 - path-failover** Configure path failover by adding a path failover target address to an interface (0 or 1).
 - ping** Check for a response from a remote IP address.
 - pwd** Display the current working directory on the screen.
-

R

- reboot** Reset the system.
- reboot force** Force the reboot without asking for confirmation.
- remove dns server** Remove a DNS server from the system.
- remove failover-config** Remove the failover configuration. This command must be followed by the `commit` command to remove the failover related commands saved previously.
- remove path-failover** Remove the path failover interface.
- remove rule** Remove a rule.
- remove server** Removes a server from a specified service or from all services in the system.
- remove service** Remove a service.
- remove service lb-group** Remove a load balancing group.

remove service lb-group rule	Remove a load balancing group rule from a service.
remove service lb-group server	Remove a load balancing group server.
remove service point	Remove a service point.
remove service ssl	Remove an SSL service.
remove service tracking	Remove tracking a service.

S

service app-monitor	Configures a service for application monitoring.
service app-monitor-param	Configures the application monitoring parameters for a service.
service lb-group	Configures load balancing group for a service, assigns a rule, and adds a required load balancing scheme. The available load balancing schemes are as follows: round-robin, weighted round-robin, static, weighted least connections, and response time.
service lb-group default	Configures a default load balancing group for a service and adds a required load balancing scheme. The available load balancing schemes are as follows: round-robin, weighted round-robin, static, weighted least connections, and response time.
service point	Set up a service point.
service qos	Set up a qos service.
service ssl	Set up an ssl service.
service tcp-dos-params	Set up a tcp-dos-params service.
service tcp-handoff-params	Set up a tcp-handoff-params service.
service tcp-params	Set up a tcp-params service.
service tracking	Set up tracking for a service.
service vlan	Set up a VLAN service.

show	Display commands.
show arp	Display all the entries in the ARP table.
show build status	Display the status of the build.
show compare-config	Compare the configuration in running memory with its correspondent configuration saved in the Flash File System. This helps you determine if the configuration has been changed and if the need to save the configuration is required.
show configuration	List all of the blade configurations as the collective output from commands: <code>show network</code> , <code>show service</code> , <code>show server</code> , <code>show rule</code> and <code>show vip</code> .
show date	Display the current system date and time.
show default tcp	Display default TCP protocol.
show end-points	Lists all the end points in the system. That is it lists all the 3 tuples and their type, for example, tracking end points, service end points, and so on.
show failover	List the failover configurations and the current failover status information.
show last build status	Display the previous build status.
show network	Display the network.
show rule	Displays the load balancing rules.
show running-config	Display the configuration that is in the running memory.
show saved-config	Show the configuration saved in the Flash File System with the option of 1 or 2. 1 indicates <code>config_1</code> and 2 indicates <code>config_2</code> .
show server	Display list of servers configured for all services in the load balancer.
show service lb-group	Display the service defined for the load balancing group.
show ssl	Display the security blade.
show system	Display the current system settings.
show uptime	Display the uptime for the system.
show vip	Display all VIPs configured in the load balancer.
shutdown	Perform a graceful shutdown of the system.
shutdown force	Force the shutdown without asking for confirmation.
ssl	Configure an SSL device.
ssl port-pair	Configure an ssl port pair.

stty columns	Set screen height.
stty rows	Set screen width.
stty hardwrap	Turns on hardwrap on the console.
stty status	Displays the console settings.

T

tar	Utility that collapses all the files under the specified directory into one tar file.
tarinfo	Displays the contents of the specified tar file.
tree	Display all the content load balancing blade commands.

U

untar	Utility that extracts all the files from the specified tar file into the current directory.
update image	Download a new boot image over the network and write it into flash.
user add	Add a new user to the system.
user delete	Delete a user.
user show	List all users currently existing in the system, along with their respective access level.

V

vip-broadcast	Configure the VIP broadcast.
vip-netmask	Configure the VIP netmask.

W

who Display all current users.

whoami Display current user, mode, and system information.

Scripting

This appendix provides a scripting example. The supported scripting currently only includes TCP connections and ASCII string based protocols like HTTP that run on top of TCP. There are four supported commands. In a script file, up to 5 connections can be established and closed. Also these connections can be intertwined. Lines beginning with # are treated as comments. The supported commands are as follows:

- `connect` – Establishes a TCP connection to the server on a specified port or on the load balancing services specified port. The format of the command is as follows:

```
PROMPTS*** connect name port name
```

Where *name* is a unique name assigned to a given connection. *name* is used with the other commands to identify the connection. *port* is the TCP port on which the connection has to be made to the server. If *port* is zero, the load balancing service's TCP port is used.

- `close` – Closes a connection established with the `connect` command. The format of the command is as follows:

```
close name name
```

Where *name* is the name of the connection to close, and is the name used when the connection was opened with the `connect` command.

- `send` – Transmits the specified data to the server on a named connection.

```
send name "data" name
```

Where *name* is the name of the connection opened with the `connect` command. *data* is the data to be transmitted to the server. It starts just after the name and has to be within double quotes. The data starts with a double quote and can span

multiple lines until the corresponding end double quote is found. If a double quote is part of the data, it has to be escaped with a `\"` character. The `\"` character is an escape character, so to include this character in the data, it also has to be escaped with the following: `\\`. `\\n`. The following character rules also apply:

- `\\n` – New line character
- `\\` – Carriage return
- `\"` – Include double quote
- `\\` – Include a back slash
- `expect` – Waits to receive a certain pattern from the server.

```
expect name "data" name
```

Where `name` is the name of the connection opened with the `connect` command. `data` is the data pattern for which to wait. The rules for `data` are the same as the `send` command except that it has one more restriction that it cannot exceed 79 characters in length.

The following codeboxes give examples of these commands:

```
#
# This example uses a http connection.
# It uses the port specified with the service VIP.
#
connect stream1 0
send stream1 "GET /index.html HTTP\\n
\\n
"
expect stream1 " 200 OK"
close stream1
```

```
#
# This example uses a 2 http connections in sequence.
# It uses the port specified with the service VIP.
#
connect stream1 0
send stream1 "GET /index.html HTTP\n
\n
"
expect stream1 " 200 OK"
close stream1
# Open a second stream
connect stream2 0
send stream2 "GET /images/image.gif HTTP\n
\n
"
expect stream2 " 200 OK"
close stream2
```

```
#
# This example uses a 2 http connections intertwined.
# It uses the port specified with the service VIP.
#
connect stream1 0
# Open the second connection
connect stream2 0
send stream1 "GET /index.html HTTP\n
\n
"
send stream2 "GET /images/image.gif HTTP\n
\n
"
expect stream1 " 200 OK"
expect stream2 " 200 OK"
close stream1
close stream2
```


Load Balancing Terms

This appendix defines some common load balancing terms, and provides examples of their use.

Sun Fire B10n Load Balancing Terms

Load Balancing Service

Defined by the destination 3-tuple, that is, the destination VIP, port, and protocol.

Example: 110.10.10.1:80:TCP

- Can be load balanced either at Layers 4 or 7
- Needs to be bound to one of the 2 interfaces on the blade
- Can be configured to support SSL if using a configuration with an SSL proxy
- When created, contains a default load balancing group with no servers or rules
- Load balancing groups with associated rules and schemes can be added
- Other attributes:
 - IP persistence
 - Cookie persistence
 - Tracking
 - Additional service access points (multi-homed service)

Load Balancing Group

- Contains a list of active servers (at least one)
- Contains a list of standby servers (optional)
- At least one rule must be specified (except for default group)
- Can add more rules or delete rules at run time
- Can add more servers or delete servers at run time
- Must have load balancing scheme specified:

- Round Robin (RR)
- Weighted Round Robin
- Static Load Balancing

Load Balancing Rule

- A rule is associated with a load balancing group in a service
- Four types of rules:
 - Hypertext Transport Protocol (HTTP) URL rule
Examples: *.html, /subdir/*, /subdir/*.html
 - CGI rule
Example: Server=MACHINE1
 - Cookie rule
Example: L7server=server1
 - IP rule
Example: 129.47.29.0:2333/255.255.255.0:0

Glossary

- ARP** The Address Resolution Protocol (ARP) conceptually exists between the data link and Internet layers. ARP assists IP in directing datagrams to the appropriate receiving host by mapping Ethernet addresses (48 bits long) to known IP addresses (32 bits long).
- Bandwidth** The difference between the highest and lowest frequencies available for network signals. Also synonymous with wire speed, the actual speed of the data transmission along the cable.
- Blade** A single board computer associated with an enclosure system that allows multiple blades to be housed in a standard server subrack, or enclosure, sharing resources such as power supplies and cooling fans. The server blade architecture is designed for computing density using a modular architecture for flexibility and scalability.
- BSC** The Blade Support Chip is an H8 micro-controller that integrates a number of different communication mechanisms and provides low-level support for a number of functions.
- Ethernet** A network communication system developed and standardized by DEC, Intel, and Xerox, using baseband transmission, CSMA/CD access, logical bus topology, and coaxial cable. The successor IEEE 802.3 standard provides for integration into the OSI model and extends the physical layer and media with repeaters and implementations that operate on fiber, thin coax, and twisted-pair cable.
- Fast Ethernet** A 100 Mbps network communication system based on Ethernet and the CSMA/CD access method.
- Full Duplex** Transmission method that allows switch and network card to transmit and receive concurrently, effectively doubling the bandwidth of that link.
- FTP** The File Transfer Protocol (FTP) transfers files to and from a remote network. The protocol includes the ftp command (local machine) and the in.ftpd daemon (remote machine). FTP enables a user to specify the name of the remote host and file transfer command options on the local host's command line. The in.ftpd daemon on the remote host then handles the requests from the local

host. Unlike rcp, ftp works even when the remote computer does not run a UNIX-based operating system. A user must log in to the remote computer to make an ftp connection unless it has been set up to allow anonymous FTP.

Gigabit Ethernet A 1000 Mbps network communication system based on Ethernet and the CSMA/CD access method.

ICMP Internet Control Message Protocol (ICMP) is the protocol responsible for detecting network error conditions and reporting on them. ICMP reports on: dropped packets (when packets are arriving too fast to be processed); connectivity failure (when destination host cannot be reached); redirection (which tells a sending host to use another router).

IPMP IP multipathing is a network load spreading and failover framework for Solaris hosts with multiple network interfaces connected to the same IP link.

Local Area Network

(LAN) A group of interconnected computer and support devices

Layer 4 The transport layer in the ISO 7-Layer Data Communications Protocol. This layer manages the transfer of data and assures that received and transmitted data are identical.

Layer 7 The application layer in the ISO 7-Layer Data Communications Protocol. This layer consists of standard communication services and applications that everyone can use.

LED Light emitting diode used for monitoring a device or network condition.

Load Balancing Service

This is defined by the destination 3-tuple consisting of the destination Virtual IP, port and protocol at which a particular service is offered. The service can be configured to load balance at Layer 4 or at Layer 7.

Load Balancing Group

This is defined by a group of back end servers, a load balancing scheme and an explicit or implicit load balancing rule. Depending on the rule matched by a connection for a load balancing service, the connection is routed to a particular load balancing group within the service and load balanced among the group of servers using the associated load balancing scheme.

Load Balancing Rule

This is associated with a load balancing group. It decides where a connection should be load balanced to. It is a description of a pattern that the connection can be matched up with. The different types of rules supported are - URL, CGI, Cookie and IP rule.

Multi-homed host In the internet environment, a single machine connected to multiple data links, which may be on different networks.

N1	Sun's vision, architecture, and products for the next-generation data center as to how to aggregate and automate distributed resources including provisioning and virtualization.
NPU	Network Processor Unit. All software on the NPU deals with packet processing in the data path.
Round robin	A load balancing scheme that distributes incoming data, using each network interface in turn.
Static Load Balancing	A scheme for distributing requests to servers where the server selection is based on static attributes of the requesting client, for example geographic location, IP addresses, and so on.
Sun Fire B1600	A platform for horizontally scalable applications that do not depend upon symmetric multi-processing to provide increased performance in a multiprocessor environment. Horizontally scaled systems achieve higher performance by dividing the application load from the clients between replicated application instances each running its own server. Each blade is a server meeting this requirement running its own instance of the operating system plus its own network application.
Sun ONE	Sun Open Net Environment (Sun ONE) is an open framework that supports web services today and lays the foundation for the smart web services of tomorrow. Sun ONE enables organizations to create, assemble, and deploy smart web services.
Telnet	The Telnet protocol enables terminals and terminal-oriented processes to communicate on a network running TCP/IP.
TFTP	The trivial file transfer protocol (tftp) provides functions similar to ftp, but it does not establish ftp's interactive connection. As a result, users cannot list the contents of a directory or change directories. This means that a user must know the full name of the file to be copied. The tftp(1) man page describes the tftp command set.
Virtual LAN (VLAN)	A Virtual LAN is a collection of network nodes that are the same broadcast domain regardless of their physical location or connection point in the network. A VLAN serves a logical workgroup with no physical barriers and allows users to share information and resources as though located on the same LAN.
Weighted round robin	A method of distributing incoming data that allows administrators to assign how much traffic a server can handle relative to the other servers in the same group. Administrators can choose these weights based on server characteristics, such as number of CPUs or CPU speed.

Index

Numerics

10/100/1000BASE-T Data Network Port
Pinouts, 28

B

build rules command, 97

C

config default tcp-params command, 68
config http-rule command, 95
config ip interface command, 52
config ip-rule command, 93
config server-monitor, 58, 59
config service lb-group default command, 70
config ssl name command, 61
create a load balancing group, 98

D

data vlan, 161
data vlan command, 160
default gateway, 55
default tcp-handoff-params command, 69
default vlan command, 161

E

enable server command, 91
enable service command, 90
enable ssl name command, 66
enable vlan command, 160
export config command, 124

H

hardware and software requirements, 3
hardware installation, 21
help command, 139

I

import config command, 125
IP address
 configuring, 52

L

LED status codes, 26
login, 49
logout, 49

M

management vlan, 161
management vlan command, 160

N

no build rules command, 98
no enable server command, 92
no enable service command, 90
no enable vlan command, 162

P

ports
 location of, 26
power off a blade, 32
power on a blade, 30

R

- remove rule command, 96
- remove service cookie-persist, 89
- remove service lb-group command, 108
- remove service lb-group rule command, 103
- remove service lb-group server command, 105
- remove service name command, 91
- remove service point, 82
- remove ssl if command, 66
- remove ssl port-pair command, 64

S

- serial port
 - pin numbers, 29
- server-monitor command, 58, 59
- service, 70
- service cookie-persist command, 88
- service ip-persist command, 83
- service lb-group command, 98
- service lb-group default command, 77
- service lb-group rule command, 101
- service lb-group server command, 103
- service point command, 81
- service ssl command, 73
- service tcp-handoff-params command, 80
- service tracking command, 85
- show build rules command, 97
- show default tcp command, 70
- show last build status command, 98
- show rule command, 110
- show server command, 111
- show service command, 109
- show service-lb-group command, 111
- shut down
 - orderly, 32
- shutdown
 - forcing, 32
- ssl port-pair command, 63, 65
- standby mode, 33

T

- TFTP, 167
 - setting up a TFTP server, 168

V

- VLANs, 11