Magnia® SG20 General Customization Guide

Contacting Toshiba

If you need assistance:

❖ support.toshiba.com

Download the latest drivers, view detailed installation instructions, and access the latest server information

❖ InTouch™ Center
Calling within the United States (800) 457-7777
Calling from outside the United States (949) 859-4273

Copyright

Disclaimer

The information contained in this manual, including but not limited to any instructions, descriptions, and product specifications, is subject to change without notice.

TOSHIBA CORPORATION AND TOSHIBA AMERICA INFORMATION SYSTEMS, INC. (TOSHIBA) PROVIDE NO WARRANTY WITH REGARD TO THIS MANUAL OR ANY OTHER INFORMATION CONTAINED HEREIN AND HEREBY EXPRESSLY DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE WITH REGARD TO ANY OF THE FOREGOING. TOSHIBA ASSUMES NO LIABILITY FOR ANY DAMAGES INCURRED DIRECTLY OR INDIRECTLY FROM ANY TECHNICAL OR TYPOGRAPHICAL ERRORS OR OMISSIONS CONTAINED HEREIN. IN NO EVENT SHALL TOSHIBA BE LIABLE FOR ANY INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR EXEMPLARY DAMAGES, WHETHER BASED ON TORT, CONTRACT, OR OTHERWISE, ARISING OUT OF OR IN CONNECTION WITH THIS MANUAL OR ANY OTHER INFORMATION CONTAINED HEREIN OR THE USE THEREOF.

Some states do not allow the exclusion of incidental or consequential damages so the above limitation or exclusion may not apply to you.

Trademarks

Magnia® is a registered trademark and InTouch™ is a trademark of Toshiba America Information Systems, Inc.

Acrobat® Reader Copyright © 1987-2001 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, Acrobat, and the Acrobat logo are trademarks of Adobe Systems Incorporated.

Microsoft, its logos, MS-DOS, Windows, Windows XP, Windows 2000, and Windows NT, and MS are registered trademarks of Microsoft Corporation.

Other product names and trademarks belong to the individual companies concerned.

About This Guide

This guide introduces the internal design of the Magnia® SG20 appliance server and provides information on how you might integrate your own software applications into a Magnia® SG20 preinstall image. Before using your Toshiba software, refer to this guide to gain an overall understanding of operating procedures.

Icons used in this guide

The following icons are used in this guide to indicate important information.

NOTE: This icon indicates information that relates to the safe operation of the equipment or related items.

CAUTION: This icon indicates the existence of a hazard that could result in the damage to equipment or property if the safety instruction is not observed.

Contents

Chapter 1 Customization

Introduction

The Toshiba Magnia SG20 is a multifunction server appliance, designed to provide ease of use and a friendly setup and management environment. The appliance is based on Red Hat Linux 7.3. Because the Magnia SG20 is an appliance, it contains the Red Hat 7.3 RPMs needed to provide its feature set, as well as a variety of tools that allow direct access and modification of the system for more experienced users. It does not contain the entire suite of Red Hat 7.3 RPMs.

The Magnia SG20 provides a number of features targeted for small workgroups, remote offices and small business. These features are configurable through its administrative Web interface, and include:

- Second HDD option (customer installable)
- Built-in 7-port local LAN switch
- Built-in wireless access point option
- Built-in modem for dial-in networking and dial-out to Internet
- Ethernet port for broadband Internet connections
- Parallel port for printer sharing
- LCD panel to display status (configurable)
- Soft power shutdown
- Easy-to-use client setup CD
- System reports
- File sharing (SMB)
- Local email
- Internet email retrieval service
- Internet gateway (sharing of a single Internet port, dialup or broadband)
- Firewall

- Internet caching and content filtering
- Backup and restore to clients or Internet
- Primary HDD snapshot to second HDD (optional)
- HDD used for additional space (optional)
- User accounts and security
- Disk quotas
- Software upgrade process over the Internet
- Internal hardware health monitoring
- Built-in Intranet (customizable)
- Easy-to-use VPN for remote access
- FTP, Telnet
- Digital Central Web site with picture, music sharing, and camera monitoring
- Home automation controller interface

When customizing a Magnia SG20 image to create a new preinstall image for your customers, it is important to ensure that the preset factory configuration is not modified without care. Doing so can result in a custom image shipped from the Toshiba factory to your end users with modifications which may result in increased service calls and a reduced initial customer satisfaction with the new system. This guide outlines some basic areas of which you should be aware, and provides some techniques for customizing the system while maintaining a favorable customer out-of-box experience.

The Magnia SG20 uses a custom Web-based interface and a set of middleware tools to enable the end user to manage the system. The internal mechanisms of this middleware use unique methods to manage system configurations, such as template-based configuration files and triggers. Direct modification of configuration files is not recommended. This guide contains information about the internal management techniques and how you might take advantage of them during your application integration process.

Customization overview

Before beginning the process of customizing a Magnia SG20 image, you can capture the clean, unmodified image on another hard disk. Doing so enables you to reinstall a factory-fresh image if you need to restart the integration process. See Custom image backup on page 13 for more information.

Generally, the process of installing a custom application involves three basic steps:

- Identifying and installing added Linux-based software to support your integration process, such as libraries, newer versions of standard Linux software, or custom support software.
- Moving your application files onto the Magnia SG20 and installing it if necessary.
- Optionally, adding links or Web pages to the Magnia SG20 Web-based administration or preinstalled intranet pages.

Accessing the Magnia SG20

For most customization, installation and modifications access the system directly using Telnet, using a shell command prompt. You can accomplish almost any task from this environment.

Access the Magnia SG20 by performing the following steps:

1. Set up a client computer connected to a private Ethernet port. Verify you can access the Web administration pages using the URL http://192.168.1.1. When this has been verified, bring up a command prompt and run Telnet:

```
telnet 192.168.1.1
```

2. When the Linux login prompt appears, use the account “telnetuser” with the default password “toshiba”.
3. When at a shell prompt, you can switch to the root account using the **su** command.
4. When at a root command prompt, you can install, recompile or modify software for your application integration.

Copying software to the Magnia SG20

When you have access to the Magnia SG20, you will need to copy your application to the system, as well as possible additional Linux support software such as databases or other tools. You can copy new software to the Magnia SG20 by using one of the following protocols:

- FTP
- SMB file sharing access (Samba)

To copy files using FTP, invoke a command prompt from any client attached to the Magnia SG20 built-in Ethernet switch. When prompted, log in to the FTP as an anonymous user (the default Magnia SG20 configuration only allows anonymous access). Use normal FTP commands to copy files to the incoming directory. When these files are transferred, use Telnet to browse to the `/home/ftp/incoming` directory. You can then move and use these files as needed.

To copy files using file sharing, connect a client to the Magnia SG20 local network and browse the network neighborhood. The Magnia SG20 will appear in the workgroup “saworkgroup” and will have the (default) computer name of “myserver”. Navigate to the share named “public”. Copy files to this directory, then use Telnet to move and use these files within Linux as needed. The “public” share is mapped to the directory `/home/public` on the Magnia SG20.

Modifying the Magnia SG20 GUI

When your application is installed and running on the Magnia SG20, you may wish to provide additional links or methods of accessing your application Web interface. There are several ways of accomplishing this:

- Add descriptive text and a hyperlink to your own Web pages.
- Add complete Web pages to the existing Web administration or preinstalled intranet.
- Add your own intranet Web site.

Listing installed Linux packages

To determine the list of Linux packages installed on your copy of the Magnia SG20 preinstall image, use the RPM command. This will allow you to determine what revision of each package has been installed on your working system. Use the following command to list the installed RPMs:

```
rpm -qa
```

User accounts and passwords

Default accounts and passwords

The Magnia SG20 is delivered with three predefined accounts:

- applianceadmin: This account is the primary administrative account for the Magnia SG20. It has access to the entire Web Administration UI, as well as complete access to all files on the system. It cannot be deleted using the Web Administration.
- telnetuser: This account logs in to the system using Telnet. While not encouraged for the typical end user, for developers it represents an important method of access to the system.
- root: This is the standard linux root administrative account with complete system privileges.

All accounts are assigned a default password of “toshiba” when delivered from the factory. Changing this password is not recommended, because doing so can make it difficult or impossible for the end user to log in and administer the system.

NOTE: Do not change the default passwords of the system.

Initial account creation

NOTE: Do not create a new user on your Magnia SG20 using the Setup CD or the Administrative Web interface. Use the existing **applianceadmin** account to access the Web Administration or to access Network Neighborhood. Use the existing Telnet and root accounts to access the system using Telnet. Modifications to the system internals can be accomplished using these accounts.

To assure that the default password for critical privileged accounts such as **applianceadmin** are changed, the Magnia SG20 uses a specific mechanism to update the password when the server is initialized. When the user creates the first user account, the password to all three predefined accounts are changed to the password of this first account.

An example scenario is for the user to open a new Magnia SG20, turn it on, and connect a PC to the built-in network. The user then runs the Setup CD to configure the client PC. During the Setup CD setup process, the Setup CD software creates a new level 3 account on the server (for example, “jsmith”). The user also selects a password at this time (for example “maxmoney1”). This process changes the passwords for the predefined accounts to “maxmoney1”.

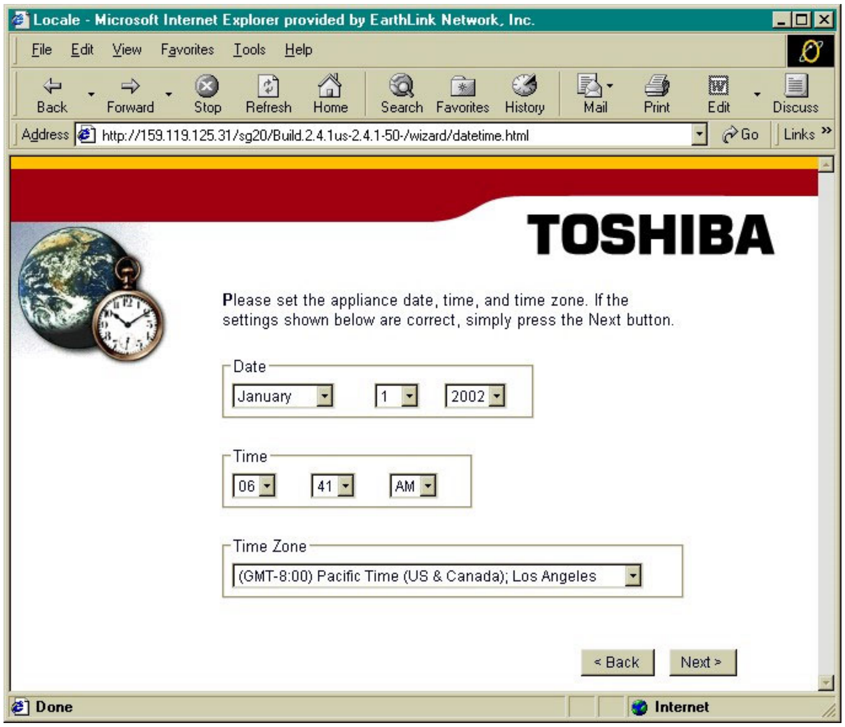
This process of assigning an initial user account password to the three predefined accounts takes place when the first user account is created. The creation of subsequent user accounts does not affect account passwords.

NOTE: This process will take place when creating an account using either the Magnia SG20 Setup CD, or using the Administrative Web interface.

Server setup wizard

NOTE: Do not run the Client Setup CD. This activates the Server Setup Wizard, and the system will no longer be flagged as uninitialized.

When the Magnia SG20 is delivered from the factory, it is internally flagged as an unconfigured, or uninitialized system. When the Setup CD is run on a client connected to the Magnia SG20, the software checks to see if the server is in its unconfigured, uninitialized state. If it is connected to an uninitialized system, the Setup CD software activates the Server Setup Wizard upon completion of the client setup.


Locale - Microsoft Internet Explorer provided by EarthLink Network, Inc.
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address
Go
Links »
TOSHIBA
< Back
Next >
Done
Internet

http://159.119.125.31/sg20/Build.2.4.1us-2.4.1-50-/wizard/datetime.html

Please set the appliance date, time, and time zone. If the settings shown below are correct, simply press the Next button.

Date: January 1 2002

Time: 06 41 AM

Time Zone: (GMT-8:00) Pacific Time (US & Canada); Los Angeles

Server Setup Wizard

When the user has completed the Server Setup Wizard, the system is flagged as initialized, and subsequently the Server Setup Wizard no longer appears.

Network configurations

NOTE: Avoid changing the networking configuration of the Magnia SG20 system while you are creating the custom image. If you require Internet or LAN access, use the “Corporate LAN or Cable Modem” option. Reset the configuration to “None” before shipping the customized image.

The Magnia SG20 is delivered with the Internet (public) connection set to “none”. This allows the user to configure the system to their personal ISP requirements (Cable, DSL, modem).

When customizing the Magnia SG20, it may be desirable to configure the system to allow Internet or corporate LAN access. This can be done, but you should be aware that after changing networking configurations, they should be returned to their original state prior to shipping an image back to Toshiba. If you do not, unintended preset networking configurations could be delivered to your end users.

If it is necessary to configure the public network for Internet or LAN access, the best method is to use the “Corporate LAN or Cable Modem” option. You can reset this option to “none” with little impact on the system.

Configuring public network access using the phone modem or DSL is less desirable. Some settings, such as phone number or user account/password, can be cached and might become part of your final customized image (an ease-of-use feature of the Magnia SG20 UI). However, this can result in the end user viewing your account (the password will be masked) if they configure modem access.

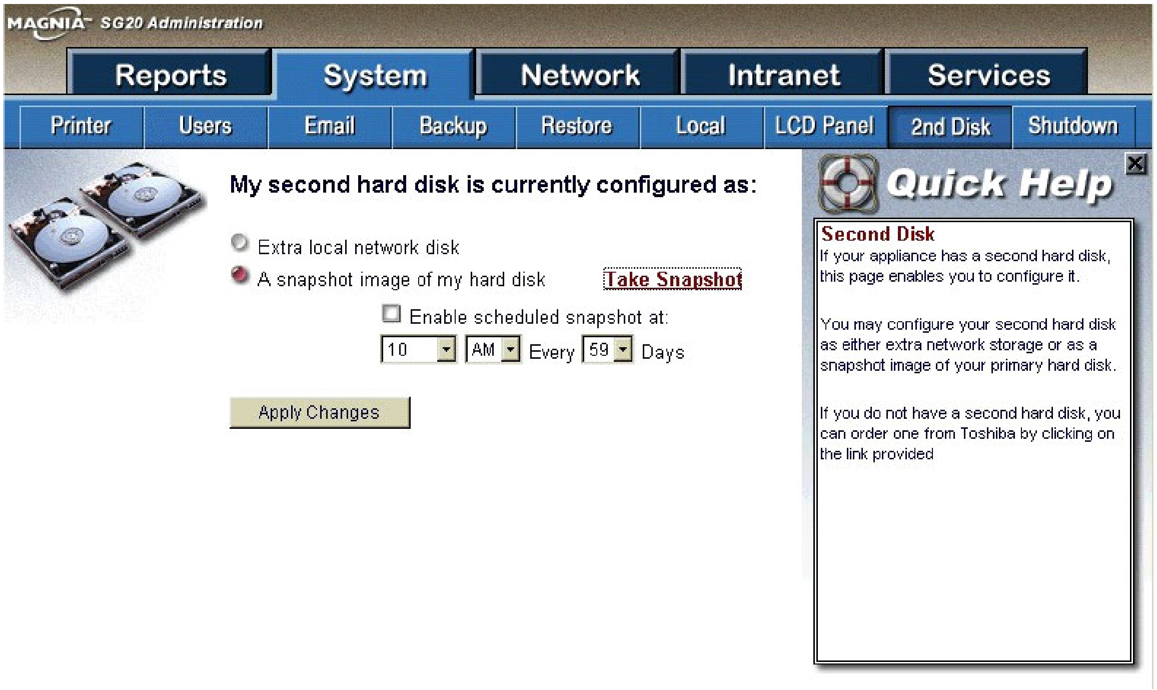
There are numerous other networking configuration options that may be changed while implementing a custom image. These can include:

- Internet caching and filtering
- Firewall settings
- High security mode
- Dial-in access
- VPN settings
- Wireless settings

These settings can be safely changed, and then reset to their factory defaults if required. You can even change these settings to your own preferred setting for shipment.

Custom image backup

When implementing the custom image, it is easy to make a mistake, requiring you to restore the system to a factory-original state, or recover it from an earlier version of your custom image. The easiest and best way to create an earlier version of your custom image is to take the original Magnia SG20 SDK preinstall image, and create a copy of it prior to doing any work using the disk snapshot feature.



My second hard disk is currently configured as:

○ Extra local network disk

● A snapshot image of my hard disk

☐ Enable scheduled snapshot at: 10 AM Every 59 Days

Second Disk

If your appliance has a second hard disk, this page enables you to configure it.

You may configure your second hard disk as either extra network storage or as a snapshot image of your primary hard disk.

If you do not have a second hard disk, you can order one from Toshiba by clicking on the link provided

Second Disk Snapshot

The disk snapshot feature takes a complete snapshot of the entire disk and places it on the secondary disk in the second slot of the Magnia SG20. Work can then proceed. If you need to go back to the original image, switch the primary and secondary disks and boot from the snapshot backup. **Do not forget to re-snapshot the drive**.

With an additional hard disk, you can also take periodic snapshots to preserve your image in case you need to return to an earlier version during your customization process.

NOTE: Take a snapshot of the original factory image before beginning work on a second disk. If needed, take periodic snapshots of your work on a third disk.

Software upgrades support

The standard retail version of the Magnia SG20 is supported by a Software Upgrades site that allows end users to view, download and apply new features and fixes. This site services customers using the standard Toshiba preinstall image only.

Because Toshiba does not control the changes or configuration of your preinstall image, each system integrator has its own Software Upgrades site. Toshiba might occasionally inform you of a software upgrade available on the standard site that you can make available to end users with your custom image. It is your decision whether Toshiba should deploy the upgrade to your customers.

To provide an Upgrade Site specific to your preinstall images, Toshiba edits a file on your image to point to the custom site before releasing the image to the factory. If you wish to retest your image after this change, or if you do not want Toshiba to make this change, inform your Toshiba sales person.

General system architecture

The Magnia SG20 is partitioned in the following manner:

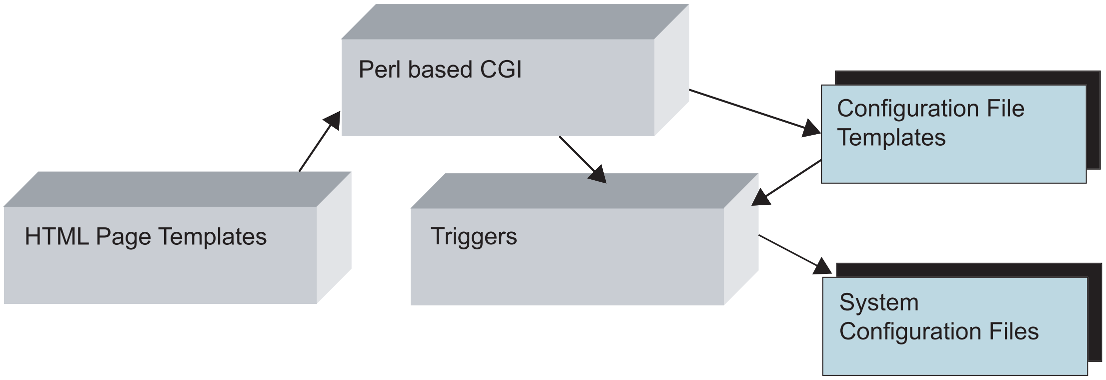
Partition	Description	Size
/	Root (system) partition	1 GB
/var	Scratch file partition, email, log files and other temporary data	1.2 GB
/home	User data	Expands to fill the remaining amount of disk.
Swap Partition	Swap area for memory pages	120 MB

The `/home` partition is where all user data is placed. It contains very little infrastructure for the support of the Magnia SG20 system. User private data and the public share, as well as user posted Intranet information are stored here. The `/home` directory contains the following subdirectories:

Directory	Description
/home/backups	This directory is a temporary location for data backup processing.
/home/docs	This directory contains documents posted for viewing using the Magnia SG20 preinstalled intranet.
/home/ftp	This directory contains the FTP applications subdirectories, such as “incoming” and “public”.

/home/intranet	This directory is a premapped location for a customized intranet. You can reach it using the URL http://myserver/intranet. You can easily create your own intranet site by placing the Web pages here.
/home/public	This directory is the open, public directory in which users can share files.
/home/users	This directory contains the individual private subdirectories for each user with an account on the system.

The Magnia SG20 system user interface, middleware and configuration are placed almost entirely in the /sa2 directory. The user interface and management general architecture is displayed in the following figure:



The Perl-based CGI uses the HTML templates, along with information from system configuration files, to create the final Web-based output for the user interface. In addition, it takes configuration changes from the user, applies them to the template-based configuration files, and invokes triggers to regenerate the system configuration files and restart any services when necessary.

- The HTML page templates contain the basic structure and graphics for the Web based user interface. These files are combined with actual data to create the final Web pages seen by the user.
- Triggers are Perl-based scripts that are executed when certain types of events occur. Triggers are called by the user interface CGI, as well as being invoked by some system callback routines.
- Template-based configuration files contain the basic information about many of the system's actual configuration files. These templates are read by triggers, and are used to regenerate the system configuration files when certain events occur.

The file system reflects the following elements in its directory organization:

Directory	Description
/sa2/templates	This directory contains the configuration file templates that are used to regenerate the system configuration files.
/sa2/lang	This directory contains the HTML and other files used to generate the actual Magnia SG20 user interface are contained in this directory. Because the Magnia SG20 can support multiple languages, this area of the system is separated into various language directory trees. For example, the “/sa2/lang/en” tree is used to support English.
/sa2/triggers	This directory contains the trigger scripts invoked to perform system reconfiguration and other maintenance.
/sa2/firewall	This directory contains definition files for special firewall rules. By adding new directories and files here, you can add your own optional firewall rules.
/sa2/conf	This directory contains all the Magnia SG20 specific configuration files. These configuration files are specific to the Magnia SG20 implementation. These configuration files do not replace the standard Linux configuration files. The primary configuration file of interest is “main.conf”, which is the central file in which most configuration items from the user interface are placed.
/sa2/lib	This directory contains Perl and other modules used as general routines to support the CGI code and other Perl programs.
/sa2/web	The Perl scripts, Java scripts, graphics and other Web-based CGI code driving the Magnia SG20. Web sites represented here are: - System Administration (/sa2/web/admin) - Preinstall Intranet (/sa2/web/intranet)
/sa2/firewall	This directory contains subdirectories for each custom firewall rule. Each subdirectory corresponds to a firewall rule the user can enable or disable through the advanced firewall settings option in the Web administration UI.
/sa2/log	This directory contains logs specific to features implemented in the Magnia SG20 user interface. The logs in this directory do not replace the standard Linux application logs typically contained in /var.
/sa2/bin	This directory contains general utility scripts and programs used by the Magnia SG20 to interface with the Linux system and the system hardware.

/sa2/upgrades	This directory contains information about the current state of the system's software upgrades. Because this area is downloaded from, and communicates with, the Toshiba Software Upgrades site, it is a good idea not to modify it. If your preinstall image is sent to the Toshiba factory, Toshiba changes a file in this directory to point to a custom software upgrade Web site for your company.

Customizing the firewall

The Magnia SG20 contains firewall based on Linux IP Tables. The default configuration for the firewall is established in the firewall template in `/sa2/templates/etc/rc.d/init.d/iptables`. In addition, there are custom firewall rules defined which the user can select or deselect. These rules are accessible through the Web-based remote administration screens in the advanced firewall page. These rules can be checked on or off (the default is off). These custom rules open holes in the firewall for items such as FTP access, Cisco VPN passthrough, and Internet gaming. Adding custom firewall rules in this area is the preferred method for customizing the Magnia SG20 firewall.

CAUTION: Modifying the firewall rules is not recommended unless you are experienced with IP Tables and firewall definitions. Adding inappropriate rules added to the firewall can cause your system to become completely inaccessible, require you to return the disk to the factory for re-imaging. Use care when establishing new firewall rules.

NOTE: Take a disk snapshot before changing or adding custom firewall rules, so that you can restore the system if needed.

Adding a custom firewall rule that can be selected and deselected using the Web administration is simple. By simply creating a new directory under the /sa2/firewall directory, and placing several simple files in this directory, a new user-selectable firewall rule is added to the user interface.

To add a new custom firewall rule, connect to the SG20 as root using Telnet, and perform the following steps.

1 Create a new directory under /sa2/firewall, using a name relating to the new rule you are adding. For example:

```
/sa2/firewall/vpnport500
```

2 In this new directory, create a new file named “rule” which contains the IP Tables command for the rule you wish to add. You can use other rule files in /sa2/firewall as examples. For example, you might add the following contents to a rule file:

```
[% IF firewall.enabled -%]

# allow VPN clients through

# Note that port 500 is the source port, not destination port.

$IPTABLES -A FORWARD -p udp -s ANY/0 --sport 500 -i $INTIF -o $EXTIF -j

ACCEPT

[% END -%]
```

3 Create a new file description in the directory, containing a short (less than one line) description of the rule to be added. This description is displayed on the user interface next to the check box. An example description is:

```
en=Custom VPN Client
```

The “en=” indicates this text description is used for any English-based display. For other languages, you can specify sp, de, it, fr, and du for Spanish, German, Italian, French and Dutch. If a language is invoked on the user interface, and there is no corresponding language text in the description file, English is used as the default.

4 Create a new file called “index” which specifies the location of the firewall within the firewall table. Location can be important, so be sure to review the other custom rules, as well as the default rules, to understand placement of the rule. The index file should simply contain a number, such as:

```
70
```

5 Finally, create a new file called “type” in the directory, containing the word “client”, "server” or “both”, that is used to describe to whom the firewall rule applies. An example type file might be:

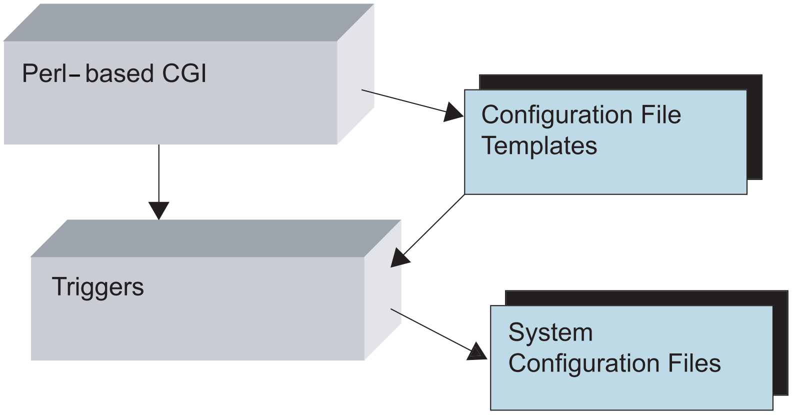
```
client
```

After these steps have been performed, a new selectable firewall rule is added to the user interface.

A very easy way to get started with a new firewall rule is to use one of the existing firewall rule directories. Copy the directory and its contents to a new directory name, and then begin modifying the individual files to change the rule display name and the IP Tables rule.

Template-based configuration files

Many of the configuration files in the Magnia SG20 are managed using templates. With this methodology, a set of templates for each configuration file is maintained. Then, when a change is made to the configuration file by the Magnia SG20 user interface or other mechanism, the configuration file is regenerated using these templates.



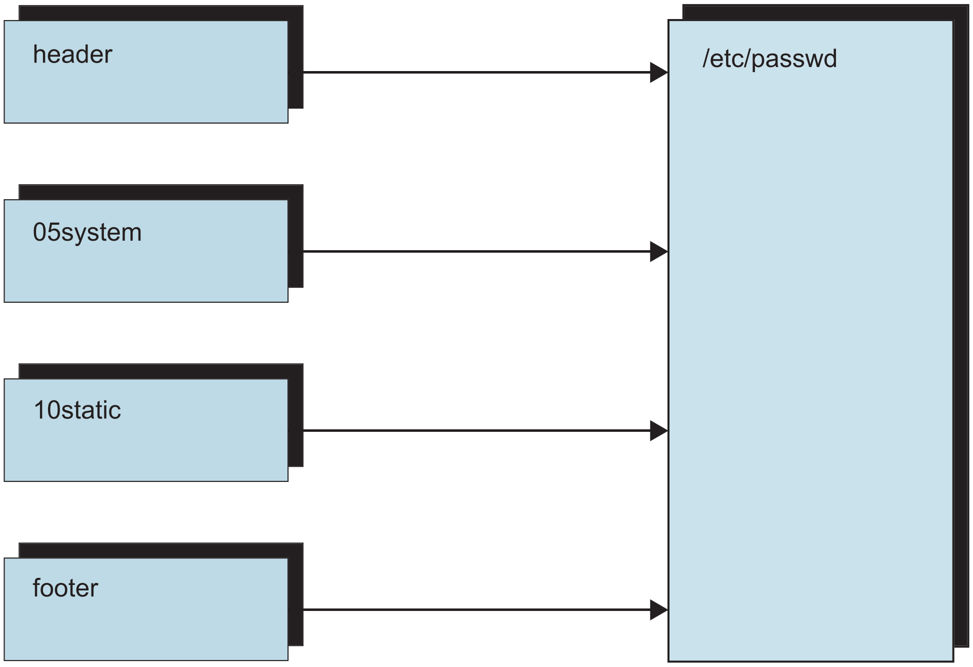
The Perl-based middleware and CGI code interfaces may need to modify the system configuration files (for example when a user is added or a configuration changed). When a change has been made, a trigger is invoked, which regenerates the corresponding system configuration file using the configuration file templates.

Because these template-based configuration files are regenerated periodically, changes to the files must be made in the templates, not in the file itself. Changing the configuration file directly does not work.

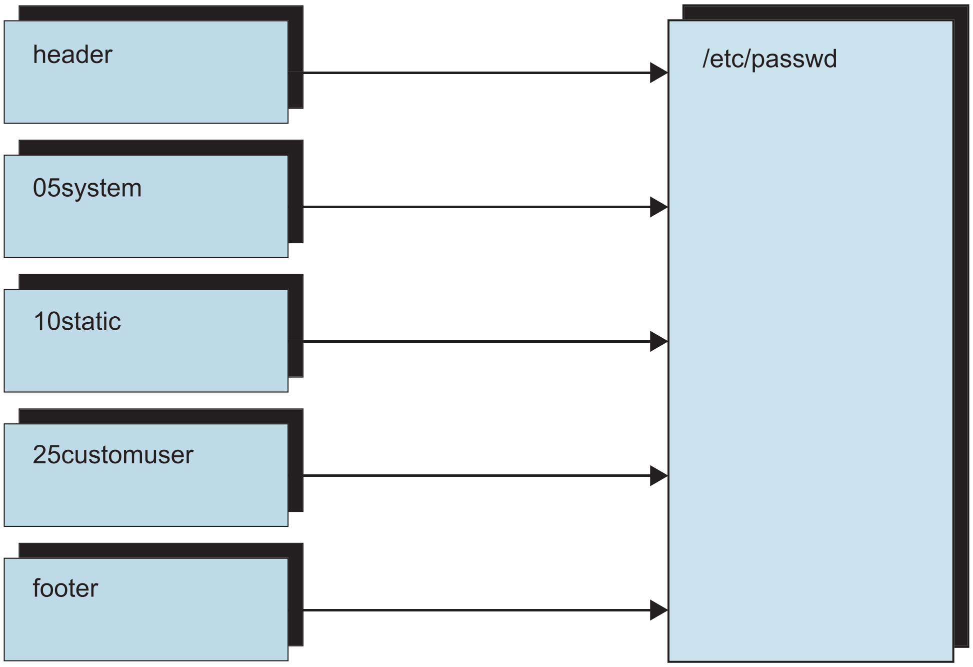
Templates are stored in `/sa2/templates`. Each file's templates are located under this directory based on the file's location in the system. For example, the `/etc/passwd` template files are located in `/sa2/templates/etc/passwd`. The contents of this directory are shown in the following table:

05system	15ftp	15postgress	footer
10static	15mysql	20dynamic	header

The configuration file may be broken up into several templates. Each template represents one part of the configuration file, and the order in which they are used is determined by the alphabetical sorting order of the template file name. This is why a two-digit number, which determines its placement within the final configuration file when it is regenerated, precedes each template file name.



To place additional elements in the configuration file, such as a new user in the password file, you can add a new template file in the configuration files template directory. Add the text desired for placement in the configuration file in a new template file. The file name should contain two leading digits that will determine your additions placement within the regenerated configuration file.



NOTE: Adding a template file does not cause the configuration file to be regenerated. Regeneration of a configuration file from its templates is only performed when the system determines that a change has been requested by the user in the user interface. In some cases, system events cause a file to be regenerated, but this is rare. For example, the password files are not regenerated unless the user adds, deletes or modifies a user account using the Web administrative user interface.

Configuration changes can be announced to the system using triggers. Triggers assure whichever actions taken are appropriate for a configuration change. Execution of a trigger will cause the appropriate configuration files to be regenerated. Please see the separate section of this document describing triggers for more detail.

The following configuration files are template-based, and should not be modified directly. This list is current at time of this document's publication. You should check your system for the actual list of files by viewing the contents of the directory /sa2/templates.

Directory	File
/etc	aliases
	crontab
	dhcpd.conf
	ftpaccess
	group
	htgroup
	htpasswd
	inittab
	named.conf
	passwd
	pptpd.conf
	printcap
	shadow
	sysctl.conf
	userdb
	wvdial
/etc/atalk	atalkd.conf
	netatalk.conf
/etc/fetchmail	fetchmail
/etc/httpd/conf	httpd.admin
	httpd.intranet
/etc/isdn	ibod.cf

/etc/mail	access
	genericstable
	local-host-names
	sendmail.mc
	virtusertable
/etc/mgetty+sendfax	login.config
	mgetty.config
/etc/ppp	chap-secrets
	ioptions
	ioptions.dialin
	pap-secrets
	pppoe.conf
/etc/ppp/peers	dialin
	pptp
/etc/rc.d/init.d	dhcpd
	iptables
	named
	network
	pcmcia
/etc/samba	smb.conf
	smbpasswd
/etc/squid	squid.conf
/etc/sysconfig	clock
	ipchains
	iptables.sh
	network
/etc/sysconfig/network-scripts	ifcfg-ethn
	ifdown-ippp
	ifdown-ppp
	ifup-ippp
	ifup-post
	ifup-ppp
/etc/sysconfig/tsb/tsb_ap_proc	channel
	distance_between_aps
	dtim_period
	enable_encryption
	enable_macfilter
	key1
	key2

	key3
	key4
	keys
	mac_accept
	multicast_rate
	network_name
	reject_any
	station_name
	transmit_key_id
/etc/xinet.d	ipop3
	telnet
	wu-ftpd
/sa2/conf	lcdkbd.conf
/sa2/web/admin	htaccess
/sa2/sbin	ifdown-local
	ifup
	ifup-local
/var/named	domain
	domain.rev
/var/spool/cron	root

Template Perl programming interface

The template processor has a Perl object-oriented interface and a command line wrapper. Each requires two parameters: filename and mode.

The usage statement of the proctmpl command-line wrapper is:

```
usage: /sa2/bin/template -f <filename> -m <octal mode> -o <owner>
```

The parameter "filename" is the name of the configuration file to be regenerated, and "octal mode" and "owner" are the file access permission modes and owner of the file being regenerated.

Output from the command-line wrapper is generated to STDOUT so that a redirect is required to overwrite the current configuration file.

Call it as shown below:

```
[/sa2/bin]#  ./template -f /etc/sysconfig/ipchains -m 700 -o

root > /etc/sysconfig/ipchains
```

The Perl object interface works like this:

```
use SA::Template;


my $tmpl = new SA::Template;


$tmpl->process('/etc/sysconfig/ipchains', '0700') or

die "Couldn't process firewall";

$tmpl->process('/etc/sysconfig/network', '0700') or

die "Couldn't process network globals";
```

This will take the template files contained in the relative path under /sa2/templates/, concatenate them, and process the concatenated value using the Template Toolkit and the values contained in the /sa2/conf/main.conf.

System startup scripts

There are two different methods to run services or scripts at boot time. The Magnia SG20 supports the standard RC boot scripts that are common on several Linux and Unix distributions. Depending on when your service or script needs to be run during the boot sequence, you can use the following methods of execution:

1. Create a standard RC script and create the proper links under /etc/rc.d/rc3.d to enable your script to be executed at a certain point in the boot.
2. Add a Perl script to the /sa2/triggers/started directory to be executed during the “started” trigger which is executed near the end of every boot. This method is not recommended for starting daemon tasks.

To start daemon or application processes at boot time, you need to make sure that they are shut down properly when an orderly shutdown is started on the SG20. You should create an RC script for starting and stopping your daemon processes. The /etc/rc.d/init.d directory on the SG20 contains several examples of scripts that stop and start daemon tasks.

A simple and straightforward example of an application start-and-stop script is the /etc/rc.d/init.d/snmpd startup script. This script starts and stops the SNMPD system daemon on the SG20. In this example, if your startup script creates an empty file in /var/lock/subsys/, then your startup script also gets called with a stop argument when the system is shutting down. It is recommended that you implement code in your start-and-stop script to handle the stop parameter and shut down your application tasks in an orderly manner. Doing so helps to avoid termination of the application tasks and loss of data when the system is shut down.

Your start and stop script should accept a "stop" parameter which will send signals to your application and daemon tasks, arranging for an orderly shutdown. Tasks that are not shut down in an orderly manner at this time are terminated by the operating system during the last stage of system shutdown.

Triggers

The internal middleware supporting the Web administration and internal operations of the Magnia SG20 are based on a trigger mechanism. Triggers are scripts (or sets of scripts) that are executed when certain events occur on the system. Though triggers may perform other tasks, the two primary tasks they perform are typically the regeneration of template-based configuration files, and the stopping and starting of various related system services. Magnia SG20 trigger scripts are written in Perl.

Events that cause triggers to be executed usually involve any detected change in the state or configuration of a specific system area or service. For example, the networking triggers may be executed when there is a change in the IP address of the public port (such as when a DHCP address is acquired). Triggers are also executed when a change is made through the Web administration user interface. For example, adding or deleting a user could execute the mod_users trigger to regenerate various password and user configuration files.

Triggers can be useful when integrating your application into the Magnia SG20 system.

- If you change configuration file templates, you can regenerate all the affected configurations and enable the change to become effective by launching the associated trigger.
- You can add your own trigger scripts to existing triggers.

The core of the trigger mechanism is a series of directories, (one per trigger, and named after the trigger), that are organized under `/sa2/triggers`. These directories may be nested as deeply as desired for organizational purposes, but the trigger name is the portion of the directory path relative to `/sa2/triggers`. Each directory contains scripts that execute actions that are tied to the trigger. The actions most likely include, but are not limited to, regenerating the template files for each dependent application, and then restarting the dependent service. The scripts in each trigger directory should be indexed by a zero-padded numeric string in a manner similar to the templates. The scripts are executed in the order that `ls -l [0-9]*` produces; therefore, the filename of each script must start with a number.

Trigger interface

The most common use of triggers in custom applications is to assure that template-based configuration changes are applied to the system after modification. In this case, executing the proper trigger from the command line is all that is needed. The usage statement of the trigger command is:

```
usage: trigger <trigger name> <args>
```

You use it like this:

```
[/sa2/bin]# ./trigger network/mod_config
```

If you are writing custom Perl code and need to execute a trigger, you can do so using the trigger object. The Perl object-oriented interface works like this:

```
use SA::Trigger;

my $trigger = new SA::Trigger;


$trigger->('network/mod_config') or

die "Couldn't trigger mod_config\n";
```

The preceding code sorts the scripts in the `/sa2/triggers/network/mod_config` directory, executes them one-by-one, waits for exit status, and returns an exit status to the caller. Trigger scripts should always return a 0 value unless there is a serious failure. If a trigger script returns anything other than a 0 value, the trigger script execution chain is stopped and the return code is passed back to the caller.

The trigger parameter correlates to the predefined trigger directory under the `/sa2/` triggers directory. Args are passed directly on the argv stack to the action scripts beneath that directory which might need to do their jobs in a more granular way:

- You might design a trigger called `network/mod_config` and pass it the name of the interface on argv so that it only has to perform the work correlating to the change of that specific interface.
- You might design a trigger called add_user and pass it the username so that the relevant configuration and password files would not have to be regenerated in their entirety; only the information relating to that user would need to be added.

However, assume that every action script could be fired by a trigger without any data and should be designed so that it regenerates all of its configuration files and restarts all services that use those files. Be very careful not to fire recursive events. Also, determine whether the service is enabled (most likely through consulting the centralized configuration file through SA:Config) and whether you need to fire given the change that was made. If the vpn only needs to reload, or if the external interface was changed and doesn't need to fire, or if the internal interface is changed, you must act accordingly to keep the execution time of a trigger to a minimum.

Manually executing a trigger

A trigger can be executed directly by executing the trigger command from a shell command line. This method can be useful when you have modified a configuration file template and want to execute the trigger that processes the template, regenerates the configuration file, and restarts any associated services. Invoke a trigger using the syntax "trigger <trigger name>". For example:

```
trigger mod_samba
```

Or

```
trigger network/mod_dialin
```

Some triggers restart networking services, which can disconnect your Telnet session.

Adding trigger scripts

You can add your own trigger to perform an application-specific action as a result of a triggering condition. For example, if your application needs to know when the public port IP address has changed, you could add a script in the network/mod_ip trigger directory. Your script, along with any other triggers in this directory, is executed when the system public IP address is changed.

Implemented Triggers

The following list of triggers is accurate at the time of the publication of this guide, and can help you understand how this mechanism is used. Verify the triggers on your system for any recent changes.

Trigger	Associated configuration files	Triggering Events
disktwo/autodetect		System boot
disktwo/crontab		UI change
disktwo/detect		System boot
disktwo/mirror		UI execution or scheduled mirror
disktwo/setconfig		UI change
disktwo/status		UI refresh
firstboot	All template-based files	Installation firstboot only
fullrestore		UI execution of full restore
mod_apache	/etc/httpd/conf/httpd.admin /etc/httpd/conf/httpd.intranet	UI change
mod_country		UI change
mod_crontab	/var/spool/cron/root	UI change
mod_email	/etc/mail/sendmail.mc /etc/fetchmail/fetchmail /etc/mail/access /etc/aliases /etc/xinet.d/ipop3 /etc/userdb /etc/mail/genericstable /etc/mail/virtualusertable	UI change
mod_firewall	/etc/sysconfig/iptables.sh /etc/samba/smb.conf	UI change, Network change
mod_health_contactinfo		UI change
mod_lcd	/sa2/conf/lcdkbd.conf	UI change
mod_lcd/locale		UI change
mod_lcd/restart		UI change
mod_samba	/etc/samba/smb.conf	Samba Workgroup change, firewall change, public IP address change if firewall is down
mod_security	/sa2/web/admin/htaccess	UI change

mod_syslang		UI change
mod_time	/etc/sysconfig/clock /var/spool/cron/root	UI change
mod_users	/etc/passwd /etc/shadow /etc/group /etc/htpasswd /etc/htgroup /etc/samba/smbpasswd /etc/ppp/pap-secrets /etc/ppp/chap-secrets	Add/Remove system user
mod_wireless	/etc/sysconfig/tsb/tsb_app_proc/channel /etc/sysconfig/tsb/tsb_app_proc/distance_between_aps /etc/sysconfig/tsb/tsb_app_proc/dtim_period /etc/sysconfig/tsb/tsb_app_proc/enable_macfilter /etc/sysconfig/tsb/tsb_app_proc/keys /etc/sysconfig/tsb/tsb_app_proc/mac_accept /etc/sysconfig/tsb/tsb_app_proc/multicast_rate /etc/sysconfig/tsb/tsb_app_proc/network_name /etc/sysconfig/tsb/tsb_app_proc/reject_any /etc/sysconfig/tsb/tsb_app_proc/station_name	UI change to wireless card settings, fullrestore trigger
modem_event		Modem status change
mondisk		Harddrive status change
monhw/hw_err		Hardware error detected
monhw/hw_limit		Hardware limit exceeded
monhw/hw_ok		Hardware status returns to OK state
monhw/hw_remote		Remote Health monitoring report status
monhw/remote_notification		Remote Health monitoring error notification
network/bring_down		Stopping of any network interface

network/enable_isdn		UI change
network/mod_dialin	/etc/ppp/ioptions.dialin /etc/ppp/pap-secrets /etc/inittab /etc/mgetty+sendfax/login.config /etc/mgetty+sendfax/mgetty.config /etc/ppp/peers/dialin	UI change
network/mod_ipsec	/etc/ipsec.conf /etc/ipsec.secrets	UI change
network/mod_proxy	/etc/squid/squid.conf	UI change
network/mod_staticip	/etc/named.conf /var/named/domain var/named/domain.rev	UI change of private IP address
network/bringup		Starting of any network interface
network/mod_config	/etc/sysctl.conf /etc/sysconfig/network /etc/sysconfig/network-scripts/ifcfg-ethn /sbin/ifup-local /sbin/ifdown-local /etc/ppp/chap-secrets /etc/ppp/pap-secrets /etc/ppp/pppoe.conf /etc/wvdial.conf /etc/ppp/ioptions /etc/sysconfig/network-scripts/ifup-ippp /etc/named.conf /var/named/domain /var/named/domain.rev /etc/rc.d/init.d/dhcpd /etc/dhcpd.conf /etc/ppp/peers/dialin /etc/atalk/config /etc/atalk/atalkd.conf /etc/ppp/ioptions.dialin /etc/inittab /etc/mgetty+sendfax/login.config /etc/mgetty+sendfax/mgetty.config	Any network setting modification
network/mod_ip	/etc/ipsec.conf /etc/ipsec.secrets /etc/named.conf /var/named/domain /var/named/domain.ver	IP address change for public port

network/mod_pptp	/etc/inittab /etc/pptp.conf /etc/ppp/peers/pptp /etc/ppp/chap-secrets	UI change to pptp VPN settings
network/mod_snmp	/etc/snmp/snmp.conf /etc/snmp/snmpd.conf	UI change to SNMP settings
remote_health		System Health Report
shutdown		Shutdown of system
started		Boot time startup scripts
started/lcdmsg		Boot time LCD initialization
starting		Boot time startup scripts
status_backup		Backup completion status
status_restore		Restore completion status
upgrades/autocheck		Auto check cron job for server upgrades
upgrades/config		UI change of upgrade autocheck settings
upgrades/current_rpms		UI execution of Upgrade check
upgrades/finish		Completion of Upgrade installation
upgrades/init		System installation – run only once.
upgrades/Newcat		New upgrade catalog received
upgrades/Start		Start of a system upgrade
upgrades/Status		Status query of system upgrade progress
users/admin_reset		Reset of admin user password

Useful files

The Magnia SG20 contains several useful files. While not all of these files can help your application integration process, they contain useful information.

File	Contents
/sa2/conf/ BUILDINFO	This file contains the current Magnia SG20 release number. This number is combined with the build version number and displayed on the LCD panel.
/sa2/ BUILDVERSION	This file contains the build number of the current release.
/sa2/conf/ main.conf	While this file contains too many fields to describe here, it is a good place to go if you want to retrieve some information about the current system configuration. Its contents are self-explanatory.
/sa2/conf/ system.conf	This file contains several items of primary interest. One is the system serial number, which is extracted and placed in this file each time the system boots. The serial number in this file is a reliable way of uniquely identifying the specific unit, for purposes such as licensing.
/sa2/ RPMVERSIONS	This file contains a list of all software packages installed on the system.

NOTE: Place your application's initials (and possibly version number) in the BUILDVERSION file. Doing so can help users and Toshiba support to identify systems running your special version of the Magnia SG20 software. Text must fit in the 16-character LCD window (along with the other build information).

LCD panel customization

The Toshiba Magnia SG20 contains a built-in, easy-to-read and use LCD panel. This display is normally used to present information to the user concerning the system status and operations. Messages include items such as:

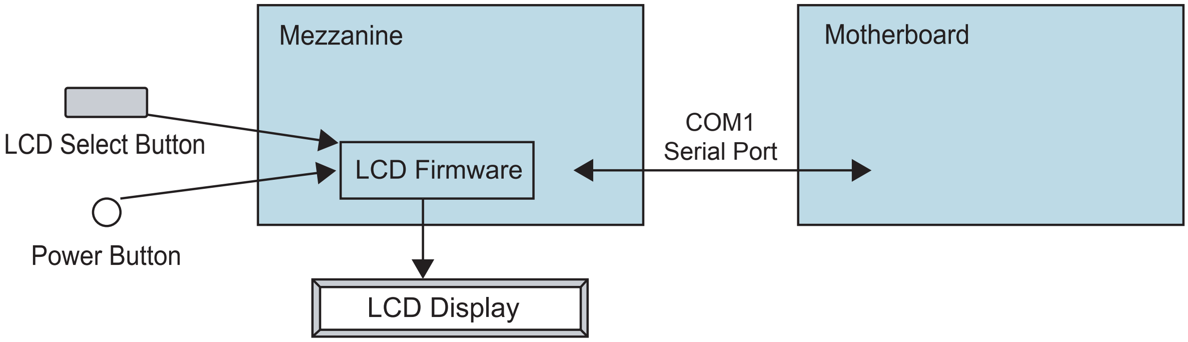
- Boot and shutdown progress
- Backup and restore progress
- Networking operations
- Health alerts

The LCD panel and the items it displays are controlled by the internal operating system of the SG20. Because the internal software controls the LCD panel, it is easy to add and modify what it displays. No modifications to the system firmware are required.

This section describes the LCD panel operation, its interface, the software controlling message display, and how to display information on the panel directly without using the Magnia SG20-provided software. There are two methods to interact with the Magnia SG20 LCD panel. When using the Magnia SG20 preinstalled software, interaction with the LCD panel is direct, through the LCD daemon. If using your own software preinstall image that does not include the Magnia SG20 software for LCD management, you can create your own programs and procedures to interact with the LCD panel and the system buttons.

LCD interface technical overview

The Magnia SG20 contains two main boards. The PC motherboard contains most of the controllers and interfaces normally associated with a PC, including the CPU, IDE controllers, serial and parallel port controllers, and so forth. A second board, the mezzanine board, contains some additional components such as the internal built-in Ethernet switch and the system LCD firmware.



Communications to the LCD display are transmitted from the operating system software running on the main system board through the COM1 serial port. The LCD controller firmware listens to the serial port and displays characters sent through this serial port.

The LCD panel is a two-line display. Each line can display up to 16 characters.

In addition to display and control of the LCD panel, the LCD firmware monitors and reports the button presses of the LCD select button and the soft power button, both located on the front of the Magnia SG20.

LCD panel direct access

It is possible to interact directly with the Magnia SG20 LCD panel without using the supplied LCD control software. This can be useful when a different operating system has been loaded on the SG20 (such as a generic Linux system, or DOS).

The techniques described in this section are for use with operating systems other than the one normally supplied with the Magnia SG20 by Toshiba. The standard Magnia SG20 software contains a Linux daemon that controls and monitors the LCD and system buttons. Therefore, attempts to control the LCD panel or to receive button-press information on the standard Magnia SG20 operating system installation conflicts with the LCD control daemon.

Direct LCD message display

Because characters sent to the LCD COM1 port are automatically displayed on the LCD screen, it is very easy to display your own messages. For example, in DOS, the command:

```
echo "Hello World" > COM1
```

causes the characters "Hello World" to appear on the LCD panel. The equivalent Linux command is:

```
echo "hello world" > /dev/ttyS0
```

More complicated strings, controlling wrapping on the two LCD lines and spacing are available. The following table shows the characters that are accepted for printing and formatting as part of the message text on the LCD panel.

Received Character	Name	Operation
‘A’-Z’, ‘a’-‘z’, ‘0-9’	ASCII printable code.	Display character at the current cursor position and advance cursor to the next location. Scroll line as needed.
SP = ‘ ‘	Space char	Advance the cursor to the next cursor position and scroll as needed.
NL := ‘\n’	New line	Advance the cursor to the next line, scrolling as needed.
CR := ‘\r’	Carriage return	Return the cursor to the first column on the current row.
ESC	Escape	Prepare for the next received char as a command code.

LCD Control Codes

In addition to direct transmission of characters to the LCD, there are several special command sequences that can be sent to the LCD controller. These sequences are designed to ease formatting and manipulation of the LCD display. All command sequences begin with the escape character, and are following by a character indicating a specific action.

The following table shows the available escape command sequences and their uses.

Received Character	Name	Operation
‘H’		Home the cursor.
‘X’		Clear the display.
‘B’		Turn on the backlight
‘b’		Turn off the backlight
<sec>‘q’		Power down the system in <sec> seconds. Should only be sent at end of the shutdown sequence.

Examples of the use of these commands appear below:

To send the LCD cursor to the top row, far left character position:

```
echo -n -e "\033H" > /dev/ttyS0
```

To clear the LCD display:

```
echo -n -e "\033X" > /dev/ttyS0
```

To turn off the backlight of the LCD display:

```
echo -n -e "\033b" > /dev/ttyS0
```

To shut the power of the system off in 10 seconds:

```
echo -n -e "\03310q" > /dev/ttyS0
```

The preceeding power-off command should not be issued unless the operating system is being shut down in an orderly manner. Sufficient time should be given for the OS to complete its shutdown procedures.

Button status codes

The LCD controller also monitors and reports the pressing of the control buttons on the Magnia SG20. Control buttons are the LCD scroll button adjacent to the LCD panel and the power-down button located on the front of the Magnia SG20 and to the right of the LCD panel.

Button status values are sent through the internal COM1 port, and are listed in the following table. No preceding characters are sent from the LCD controller to the motherboard.

Transmitted Character	Name	Operation
'P'	Power	Power button is pressed.
'S'	Select	Select button is pressed.

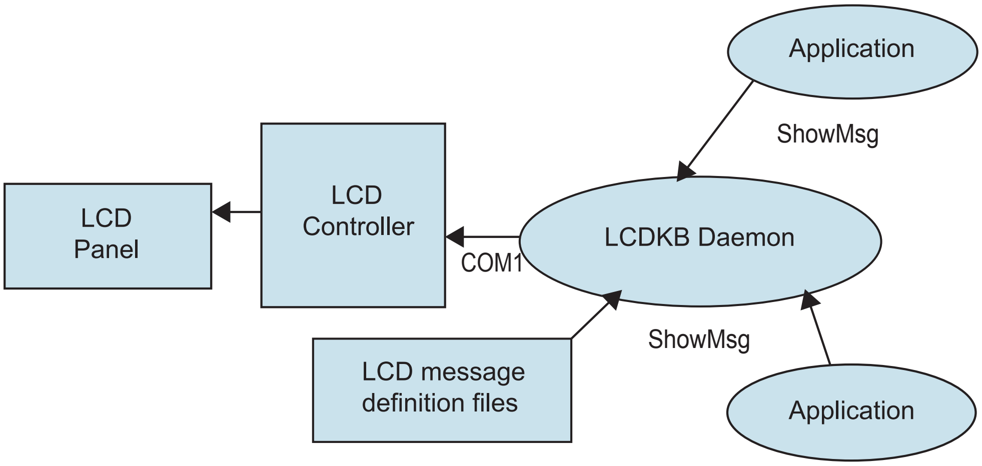
Because the preceding characters are sent through the COM1 port without a CR/LF, it may be necessary to write specific code to monitor the incoming characters. This code should place the device in a raw mode to receive characters as soon as they are sent, so that they are not buffered. While these actions can be accomplished using command line level instructions, it is better to write a program to monitor and process these commands.

While command and shell scripts can be used to send information to the LCD for display, monitoring and responding to button commands is best handled in a program specifically written for this purpose.

Magnia SG20 queued LCD message interface

To add custom LCD messages when using a Toshiba Magnia SG20 preinstall, you need to integrate your messaging into the internal messaging management system. The internal queuing system manages the display of LCD messages, their positioning, translation and cycling through the multiple messages available for display using the LCD scroll button.

An LCD/KBD daemon (LCDKBD) coordinates client application access to the LCD/KB micro controller subsystem. This is a Linux daemon that runs at system startup and is used by client applications to queue messages for display on the LCD panel. Client applications do not directly communicate to the device.



Direct command line interface

The command line interface to the LCD/KB port is used for direct display of messages on the LCD panel.

The following describes the command line interface for script access.

```
    setlcd [-<cmd> [<data>]] <text>
```

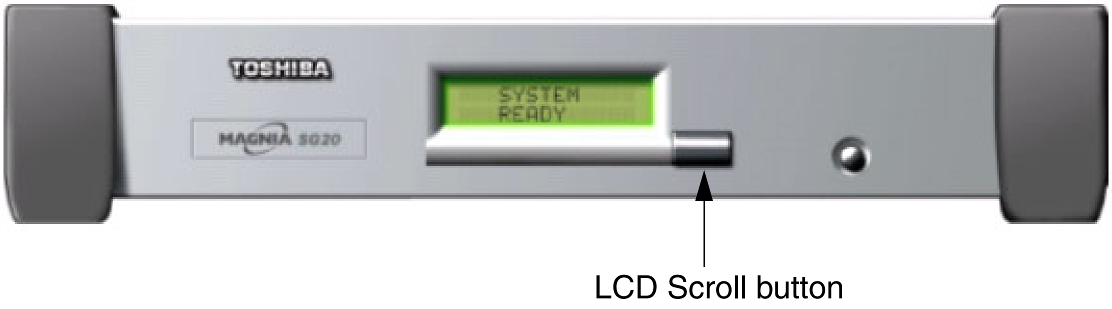
where -<text> is an ASCII text string with NL ('\n') & CR ('\r') support, and where -<cmd> is described by the following:

Cmd	Description	Comment
clear	Clear the display.	Clear the display and home the cursor.
home	Home the cursor.	Move to the row=1, col=1 position.
home2	Move the cursor to the second row.	Move to the row=2, col=1 position.
col <col>	Move the cursor to the column position.	Move the col to <col>.
row <row>	Move the row to the row position.	Move the row to < row >.

Because this command bypasses the message queuing system, it is reserved for use by the LCDKBD and for debugging.

Queued LCD messages

Under normal circumstances, all LCD messages are displayed and managed by the LCDKBD daemon. This daemon takes messages that should be displayed on the LCD panel and places them in a queue. Because multiple messages are available for display on the LCD panel, the LCD scroll button on the front of the machine is used to communicate with the LCDKBD daemon.



LCD Scroll button

When the LCDKBD daemon detects the press of a button, it cycles through the messages in the message queue. Some messages can be forced to the top of the queue, placing the message on the LCD without requiring that the user press the LCD button. This technique is used for urgent messages such as alerts. Messages can also be displayed for one time only. This message appears on the LCD until the user presses the LCD scroll button, which removes the message from the message queue so that it does not appear again.

Use the Perl object ShowMsg to tell the LCDKBD daemon to display specific messages on the LCD panel, as described in the following section.

Adding LCD messages

Because the contents of the LCD display are managed by the LCDKBD daemon, display of new messages involves creating a new message definition in an LCD message definition file. When defined properly, the LCD message can be displayed on the LCD panel by sending the LCDKBD daemon an instruction. Use the following procedure to add your own LCD message to the display message queue.

1. Go to the directory /sa2/lang/en/lcdmsg. This directory contains all LCD message definition files for the English language (the default). When customizing messages to display in different languages, go to the corresponding language directory, such as /sa2/lang/es/lcdmsg for Spanish, or /sa2/lang/de/lcdmsg for German.

2. Create a new definition file for your LCD message. An example of a new message definition file called “newmsg” follows:

```
newmsg.desc=Acme Application Messages
newmsg.name=ACME
newmsg.readonly=0
newmsg.hidden=0
newmsg.msgup=-center “Acme App” -center “Running”
newmsg.msgdown=-center “Acme App” -center “Not Running”
newmsg.msgalert=-ontop -center “Acme App” -center “%s Error”
newmsg.msgmail=-ontop -onetime -center “Acme App” -center “Check Mail”
```

See LCD message display options on page 40 for more information about these fields and their operation.

3 Add a Perl script to your application that invokes the messages defined in your LCD message definition file. The script should incorporate the appropriate ShowMsg call to display a message. An example program follows:

```
use SA;
use SA::LCD::Disp;

my $objDisp = SA::LCD::Disp->new();
$objDisp->ShowMsg(“newmsg.msgmail”);

exit(0);
```

This script places the message “Acme App” and “Check Mail” on the first and second lines of the LCD panel.

To display an LCD message that incorporates dynamic text, such as the newmsg.msgalert LCD message in Step 2, use the following:

```
$objDisp->ShowMsg(“newmsg.msgalert”, “Download”);
```

4 To complete the configuration process for the new message, execute the mod_lcd trigger. This trigger regenerates the LCD configuration files and informs the LCDKBD daemon that it should refresh its message list.

```
trigger mod_lcd
```

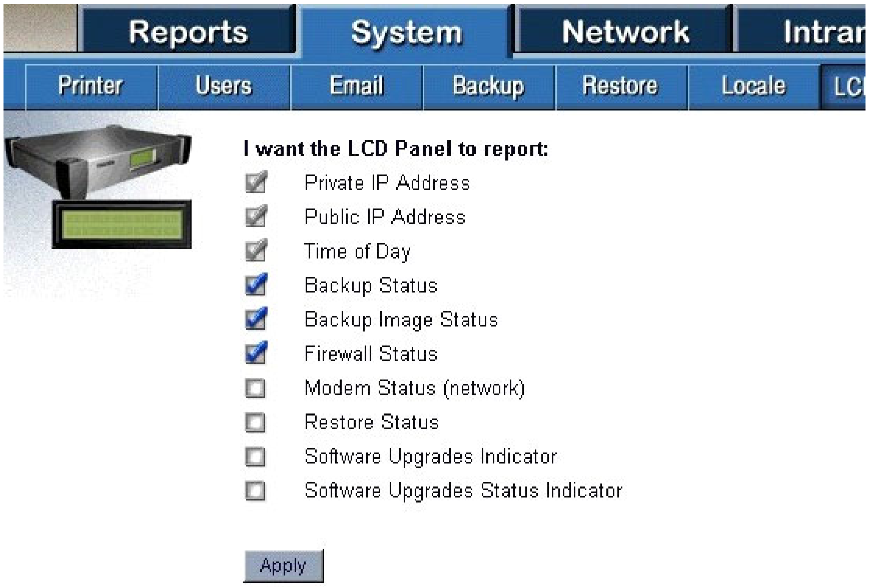
The execution of this trigger is only necessary when you have made a change to the LCD message list.

LCD message display options

The LCD message definition files contained in the /sa2/lang/en/lcdmsg directory contain a variety of options (see Step 2 of the preceding example). Fields are defined by the name of the message with a specific configuration parameter suffix.

```
newmsg.desc=Acme Application Messages
```

The “desc” portion is the configuration field parameter indicating the text description of the LCD message, and appears on the LCD configuration page in the Web administration user interface that allows the user to select which LCD messages to display.



I want the LCD Panel to report:

☑ Private IP Address
☑ Public IP Address
☑ Time of Day
☑ Backup Status
☑ Backup Image Status
☑ Firewall Status
□ Modem Status (network)
□ Restore Status
□ Software Upgrades Indicator
□ Software Upgrades Status Indicator

LCD Configuration Screen

The name suffix provides and internal name used by the Magnia SG20 middleware to identify the message being displayed or configured.

```
newmsg.name=ACME
```

The name of the message is “ACME”.

```
newmsg.readonly=0
```

The read-only field specifies whether the user is allowed to change the LCD setting. In the default LCD configuration screen, the IP address display and time of day are always checked and cannot be unchecked by the user. To allow the user to prevent this LCD message from displaying on the LCD panel, set the read-only field to 0. To prevent the user from deselecting this message and force the message to always display, set the read-only field to 0.

```
newmsg.hidden=0
```

The hidden field specifies whether the LCD message appears on the configuration screen. Some messages, such as health monitoring, do not even appear on the user LCD configuration screen, and are always displayed. To prevent a message from appearing on the configuration screen, set the hidden field to 1. To present the message on the user configuration screen, set the hidden field to 0.

```
newmsg.msgup=-center “Acme App” –center “Running”

newmsg.msgdown=-center “Acme App” –center “Not Running”

newmsg.msgalert=-ontop -center “Acme App” –center “%s Error”

newmsg.msgmail=-ontop -onetime –center “Acme App” –center “Check Mail”
```

A single LCD message can have several states, or submessages. For example, the backup message may display the backup status as Not Performed, Started or Completed. Only one of these messages can be displayed on the LCD panel at a time. If the Started message is displayed, and then later the Completed message is displayed, the Completed message replaces the Started message.

In the proceeding example, the Acme App message displays “Acme App” on the first LCD line. The second line will be Running, Not Running, Error or Check Mail.

The following formatting options are available:

- -center: Center the text in the middle of the LCD line.
- -ontop: Immediately display this message by placing it on the top of the message queue.
- -onetime: Clear this message from the LCD message queue after it has been viewed by the user once.

Reader Response Questionnaire

Toshiba welcomes your opinion. Your comments provide valuable information that helps us refine the documentation that ships with our products and evaluate the usefulness of support services that Toshiba currently offers.

A Reader Response Questionnaire and self-addressed mailer are on the following two pages.

You may also send comments to: magnia@tais.toshiba.com

Printing the Questionnaire and Mailer

To print the questionnaire and the self-addressed mailer:

1. Select **Print** from the **File** menu

   The Print dialog box displays.

2. Under **Print Range**, click the **Pages**
3. In the **from** box, type 43.
4. In the **to** box, type 44.
5. Click **Ok**.

Folding the Questionnaire and Mailer

To prepare the questionnaire for mailing, you will need to fold the questionnaire and the self-addressed mailer and then tape the two pages together.

1. Place the self-addressed mailer face down on a flat surface.
2. Place the questionnaire on top of the mailer.
3. Fold both sheets together along the lines provided, with the mailing address on the outside flap.
4. Seal the complete mailing with tape (no staples), add appropriate postage, and mail.

We Want to Hear from You

Toshiba welcomes your opinion. Your comments provide valuable information which will help us improve support services.

Toshiba Magnia SG20 General Customization Guide

Please check all that apply:

I develop using:

- □ Red Hat
- □ Windows 2000
- □ Windows XP Pro
- □ .NET
- □ other ________________________________
- □ SuSE
- □ Caldera
- □ United Linux
- □ Windows Embedded

Please answer each question below by checking the appropriate response:

1. **I think the Toshiba provided Quick Start Card would be helpful for our end user's.**
   □ strongly agree □ agree □ neutral □ disagree □ strongly disagree
2. **I found the Magnia SG20 General Customization Guide easily on the Software and Documentation CD.**
   □ strongly agree □ agree □ neutral □ disagree □ strongly disagree
3. **It was easy to access the information I needed in the electronic Magnia SG20 General Customization Guide**
   □ strongly agree □ agree □ neutral □ disagree □ strongly disagree
4. **The information was easy to read and understand.**
   □ strongly agree □ agree □ neutral □ disagree □ strongly disagree
5. **I found the Magnia SG20 General Customization Guide provided the information relative to my setup requirements.**
   □ strongly agree □ agree □ neutral □ disagree □ strongly disagree
6. **Are you planning to use Adobe Acrobat Reader to view other documents that you are supplying to your end users?**
   □ yes □ no □ not sure
7. **Would you prefer the System's User's Guide to be an HTML document versus a PDF?**
   □ yes □ no □ not sure
8. **Would you prefer to access the Internet for all system integrator documentation?**
   □ yes □ no □ not sure
9. **Are you planning on providing a printed User's Guide to your customers?**
   □ yes □ no □ not sure

ADDITIONAL COMMENTS

____

____

____

____

____

____

____

____

(Optional Information)

Name: ____
E-mail address: ______________________________ Tele: (____) ______________
Company: ___________________________________ Title: __________________
Street address: ____
City: ____________________________________ State: ________ Zip: __________
□ Yes, I would like to assist Toshiba in evaluating their computer materials in the future.

Fold here

Fold here

Toshiba America Information Systems, Inc.
Attn: Magnia Product Manager A135E
Computer Systems Division
9740 Irvine Blvd.
Irvine, CA 92618-1697