Guardian Digital WebTool Firewall HOWTO

by Pete O’Hara

Guardian Digital WebTool Firewall HOWTO

by by Pete O'Hara

Revision History

Revision $Revision: 1.1 $ $Date: 2006/01/03 17:25:17 $ Revised by: pjo

Table of Contents

List of Tables

Chapter 1. Guardian Digital WebTool Introduction

The Guardian Digital WebTool is a secure on-line administration utility accessed by using your browser. You have the capability to control every aspect of the system through the Guardian Digital WebTool utility. The Guardian Digital WebTool abstracts the difficult process of configuring a secure system, easing system setup and administration. Designed to simplify complex network administration tasks, the Guardian Digital WebTool provides the unique ability to monitor security activity, build secure web sites, and manage email and other network functions.

Chapter 2. Firewall Introduction

The Guardian Digital WebTool *"Firewall Interface*" is based upon "*Shorewall*", a user interface to the "*iptables*" utility which controls the network packet filtering code in the Linux kernel. With this the Guardian Digital WebTool *"Firewall Interface*" will provide everything necessary to setup the EnGarde Secure Linux firewall. You will be able to define zones and networks, default ACCEPT/REJECT policies for each interface, setup internal network masquerading, rules, port forwarding and blacklists.

Chapter 3. Document Goals

In our example we will configure a firewall located between two networks, external (untrusted - 192.168.1.0) and one internal (trusted - 10.0.99.0). Here is a table of the servers and their IP addresses.

Table 3-1. Server Addresses

Server	Address
Firewall (external interface)	192.168.1.81
Firewall (internal interface)	10.0.99.1
Administrator's host	192.168.1.150
Mail server	10.0.99.11
DNS server	10.0.99.12
Web server	10.0.99.13
SIMAP/SPOP3 server	10.0.99.11

As just one example, in your environment the 192.168.1.0 network used here could represent your external network or the Internet and the 10.0.99.0 network could be your internal network. In this type of scenario the firewall is made to look like a group of servers (for example DNS, SMTP, FTP, HTTP, etc.) to the external network. The Internet facing interface (represented here by 192.168.1.81) receives the packets for all of these services and then forwards them to the appropriate servers on the internal network (represented here by 10.0.99.0/24) in the private IP network space. This is different from having a group of servers in a publicly addressable DMZ. I chose this example as it involves firewall configuration that exercises more functionality and provides the reader with a broader information base. This is also useful if the user has very small group of public IP addresses. In fact this could be used when there is only one public IP address to represent a group of servers and services. In addition to this our example will accomplish the following:

- Define two networks and several hosts.
- Allow incoming Guardian Digital WebTool (port 1023) packets only from the administrator's host to the firewall and internal network.
- Allow incoming SSH packets from the administrator's host to the internal network.
- Allow outgoing HTTP(S), FTP, SSH and NTP packets from the internal network.
- Allow outgoing SMTP packets only from the internal mail server.
- Allow outgoing DNS packets only from the internal name server.
- Blacklist all incoming packets from a malicious external host.
- Block all incoming packets from the external network that contain invalid TCP flags.
- Port forward incoming SMTP requests to the mail server.
- Port forward incoming DNS requests to the name server.
- Port forward incoming SIMAP and SPOP3 requests to the mail server.
- Port forward incoming HTTP and HTTPS requests to the web server.
- Masquerade the internal network.

Note: A general note about defining networks. Networks in the Guardian Digital WebTool are defined in CIDR notation so for example the network 192.168.1.0 with a netmask of 255.255.255.0 would be defined as 192.168.1.0/24. You will need a general grasp of how CIDR notation works or you may improperly configure the firewall in certain instances. *If CIDR is new to you, you should research how it works before continuing the firewall configuration.*

Note: In the networking world there is the public and private address space. The public address space is made up of typical IP addresses that can be routed through the Internet. The private address space is made up of a group of IP addresses reserved specifically for use in private networks that are not directly connected to the Internet. Understanding these concepts is absolutely necessary before continuing the firewall configuration. *If you are not totally familiar with the distinction between public and private IP addresses you need to stop here and research these concepts.*

Chapter 4. Configuring the Firewall

4.1. General Configuration

To access the firewall configuration section in the Guardian Digital WebTool place your cursor over "*System*" and pull down to "*Firewall Configuration*" and click. Now place the cursor over "*Module*" in the title bar and there will be five categories to choose from. To choose a category place the cursor over the category and click.

4.1.1. Firewall Interfaces

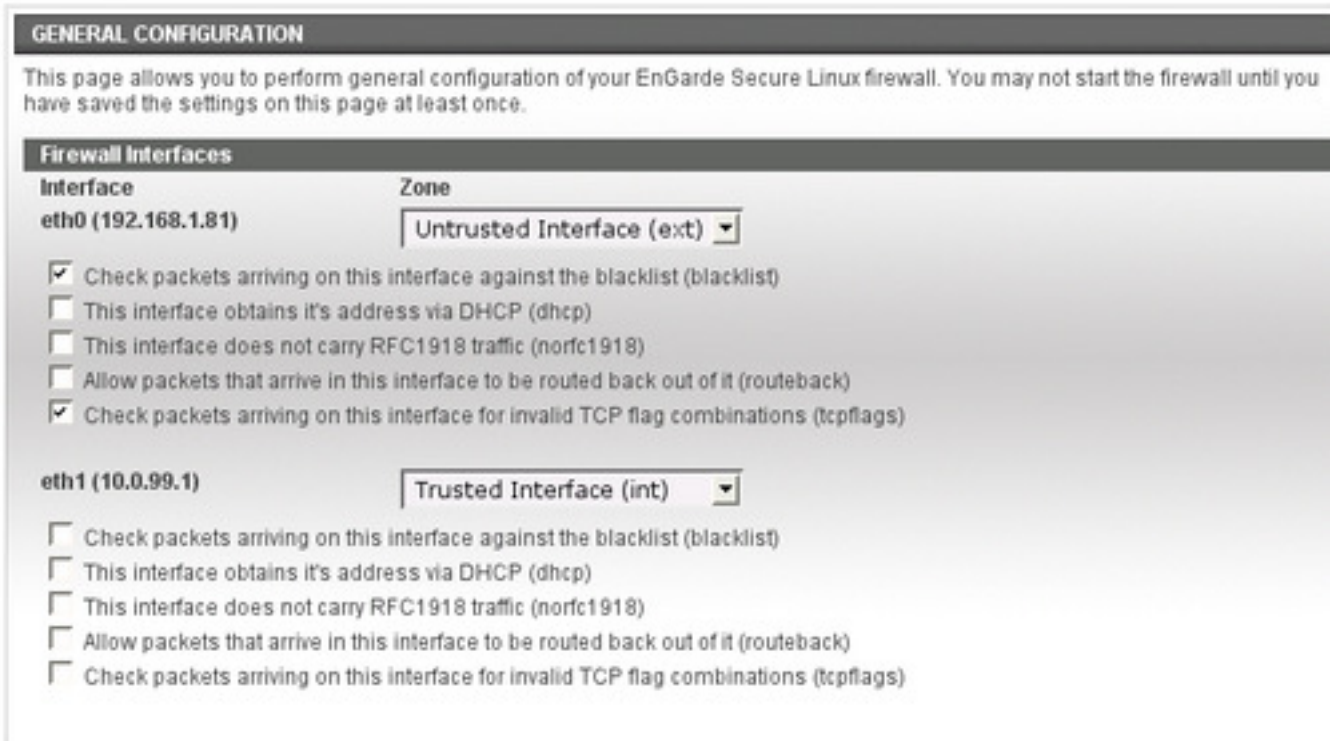
This is where the interfaces and options are defined. There is a section for each interface on the firewall. Here we define external interface eth0 as untrusted by using the pull down menu and selecting "*Untrusted Interface (ext)*". Do the same for the internal eth1 interface and select "*Trusted Interface (int)*". It's up to you whether you would like to apply the blacklist and TCP flag options to the internal interface. For the example they are not applied as hosts on this interface are considered trusted. The terms 'trusted' and 'untrusted' are self explanatory. This definition tells the software which side of the firewall is to be protected.

Since we are going to create a blacklisted machine that resides on the external interface somewhere we want to enable blacklisting on this interface. Check the blacklisting check box to do this.

It is also a good idea to block packets with invalid TCP flags coming in from the outside so we will check this check box as well.

Once you have made your selections click on "*Save Configuration*" at the bottom of the page. This only saves the configuration and does not alter the current state of the firewall. You need to restart the firewall to actually put a new configuration into place. This is done in the "*Service Configuration*" section of the Guardian Digital WebTool and will be discussed later on. If this is the initial firewall configuration you should ensure that the firewall is off (See Section 4.6 below and ensure the firewall is disabled).

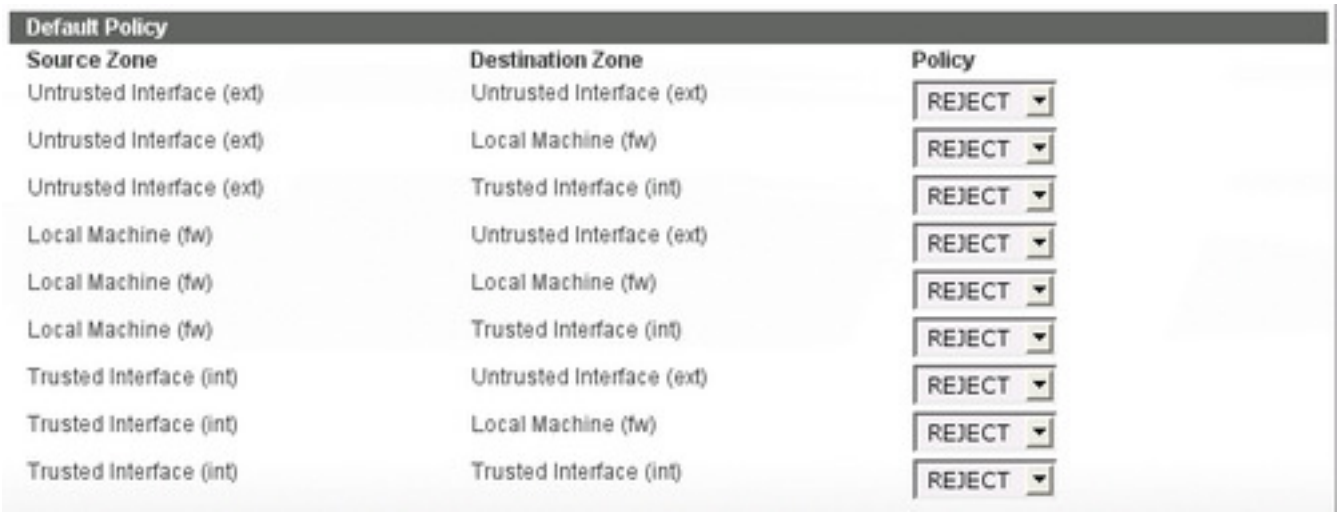
Figure 4-1. Firewall Interfaces Display



4.1.2. Default Policy

The default policy defines the default action that will be implemented by the firewall. It is the starting point upon which firewall rules (which will be created later) will build on. The two policy choices are ACCEPT or REJECT. Using ACCEPT means that by default you accept all packets sent by the '*Source Zone*' and destined for the '*Destination Zone*'. REJECT means that by default you reject all packets sent by the '*Source Zone*' and destined for the '*Destination Zone*'. For instance, in this example all of the default policies for all source/destination combinations are REJECT and is the safest choice. By using REJECT you deny all packets and later you define ACCEPT rules for the particular type of packets/services that you want to pass. This is much easier than using an ACCEPT default policy and then defining REJECT rules for every type of packet you don't want to allow to pass through the firewall.

Figure 4-2. Default Policy



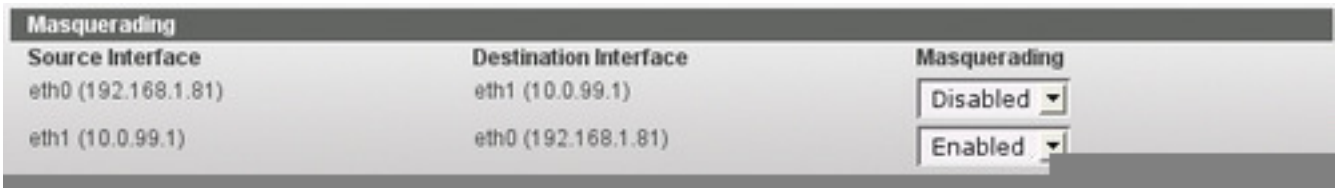
4.1.3. Masquerading

Masquerading is an action that hides the internal network from the external world. It allows the internal network to be in the private address space. One of the advantages of this is that it economizes the amount of IP addresses that are needed for the presence of an organization on the Internet. You can define thousands of internal IP addresses while only using a few public IP addresses. There are different reasons for using or not using masquerading (also known as NAT - Network Address Translation). In this example masquerading is turned on.

When this is deployed any outbound packets from the internal network and passing through the firewall will have the internal source IP address replaced with the firewall's external address. For example DNS request packets submitted from the internal DNS server (10.0.99.12) will have its source translated from 10.0.99.12 to 192.168.1.81 before leaving the external firewall interface. Now the packet has a source IP address that is in the public address space and can be routed through the Internet appropriately. The response packets from the Internet will be addressed to 192.168.1.81, the external address of the firewall. Because the firewall is masquerading the internal network it keeps a table of the outgoing packets and when it receives a response from the Internet to an internal request it will be able to restore the proper internal IP address for the destination and send it to the original internal requesting machine.

This is easily configured by enabling masquerading for the source interface of eth1 (10.0.99.1 - the internal interface) and destination eth0 (192.168.1.81 - the external interface).

Figure 4-3. Masquerading

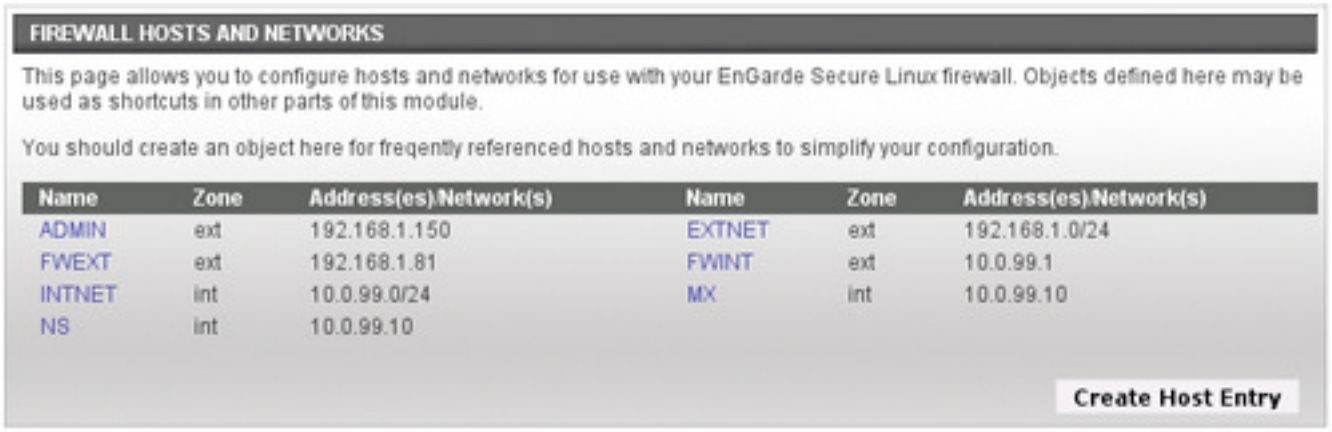


4.2. Hosts and Networks

In this section there is the ability to give hosts and networks convenient naming schemes. This will make firewall rule creation easier on us humans as these names can be used in place of entering IP and network addresses for sources and destinations. In our example scenario we have several entities that would be advantageous to name here. There are the two networks (internal and external), the two firewall interfaces (internal and external), a mail server, a DNS server and an external host where remote administration can be performed from.

- Two networks (internal and external)
- Two firewall interfaces (internal and external)
- Mail server
- DNS server
- Administrator's remote host

Figure 4-4. Hosts and Networks



4.3. Firewall Rules

In this section the firewall rules are created. *There are two types of rules, ACCEPT and REJECT*. In ACCEPT rules you define packet types that will be accepted by the firewall and REJECT rules define packet types to be reject. You can think of these rules as exceptions to the default policy. Since a REJECT default policy is being used the type of rules that will be needed are ACCEPT rules. The first step in creating firewall rules is to list the services that should be allowed with their sources and destinations. So let's take another look at what our example is to accomplish.

- Allow incoming Guardian Digital WebTool (port 1023) packets from the administrator's host to the firewall and internal network.
- Allow incoming SSH packets from one specific host to the internal network.
- Allow outgoing HTTP(S), FTP, SSH and NTP packets from the internal network.
- Allow outgoing SMTP packets only from the internal mail server.
- Allow outgoing DNS packets only from the internal name server.

4.3.1. Incoming Rules

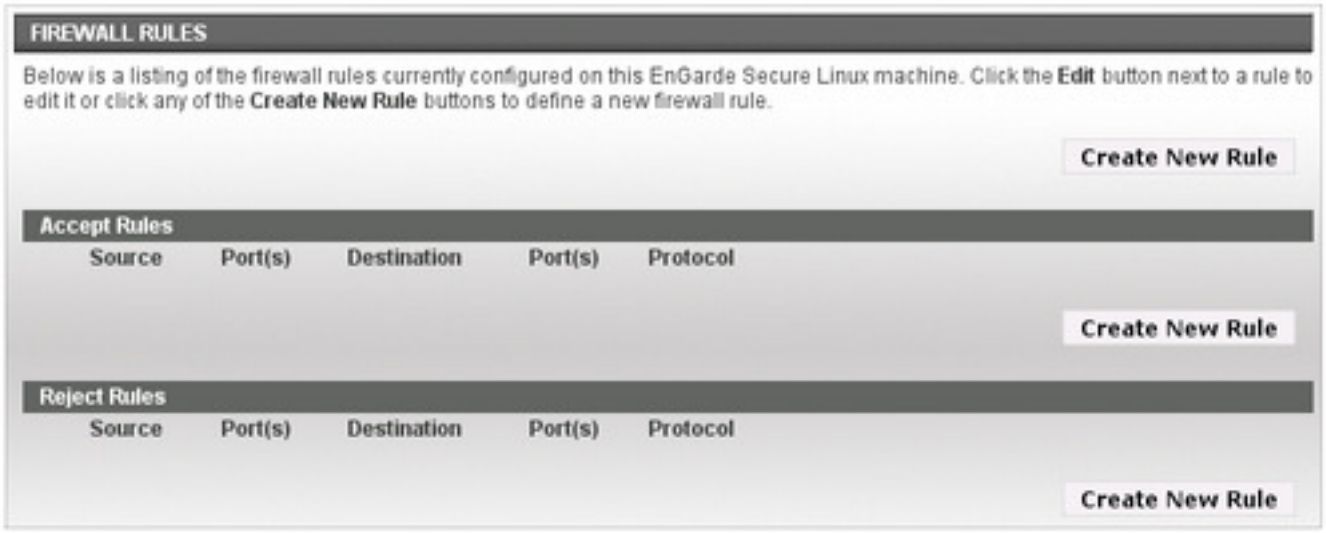
Here we are going to create the rules for incoming packets. In our example the only services that we are going to create actual rules for are the Guardian Digital WebTool and SSH. The other incoming services will be handled later on in *Port Forwarding* (See Section 4.4.

4.3.1.1. Guardian Digital Webtool - port 1023

The very first thing that you want to do is to make sure that you can connect to the firewall via the Guardian Digital WebTool. This is important because as you experiment with rule modifications you could easily lock yourself out of the firewall. If you at least have WebTool access then you can always turn the firewall off or fix the error and an re-enable access.

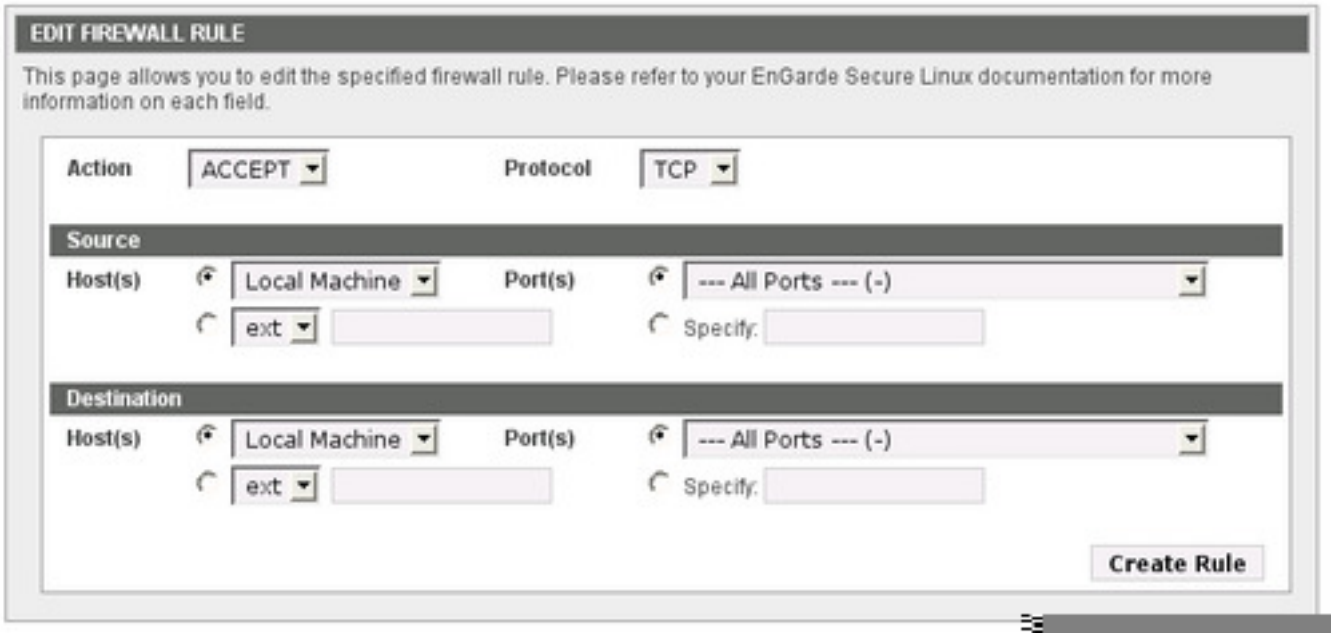
So let's create our first rule. Access the "Firewall Rules" category in the Guardian Digital WebTool. You will see a blank rule page with three sections.

Figure 4-5. Firewall Rules Page



To create a rule click on a "*Create New Rule*" button and you will see a pop-up window.

Figure 4-6. Edit Firewall Rule Pop-up



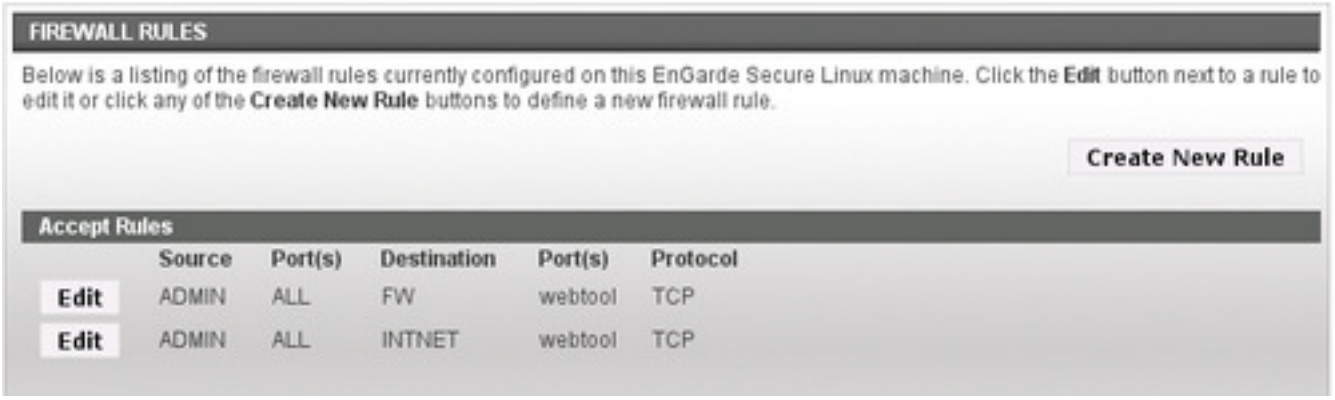
As you may recall, we defined all default policies as REJECT. So we need to create ACCEPT rules to let packets on a particular port to pass.

- Choose the ACCEPT action.
- The protocol that the Guardian Digital WebTool uses is TCP so choose this protocol. Most services use the TCP protocol. Some exceptions are DNS (which uses both TCP and UDP) and NTP which uses UDP.
- We want pass all packets from the administrator's machine to the firewall. Check the "*Host(s)*" radio button and use the pull down menu for the source "*Host(s)*" and choose "*ADMIN*".
- The source port selection needs to be "*All Ports*" as the Guardian Digital WebTool request can come from any port. (Make sure that you have selected the radio button next to the "*Port(s)*" pull down menu).
- The destination is the firewall itself so once again select the "*Host(s)*" radio button and use the pull down menu to select "*Local Machine*".
- For the destination port use the pull down menu and select "*Guardian Digital WebTool (webtool)*". The actual port number is 1023 but you don't need to be concerned with that as the Guardian Digital WebTool will use this port behind the scenes.
- Now click on the "*Create Rule*" button. The pop-up will disappear and you will now see the rule that you have just created listed under the "*ACCEPT*" rules.

To allow the administrator's host to connect to any other machine on the internal network via the Guardian Digital WebTool repeat the above with the exception of choosing the "*INTNET*" in the destination host pull down menu. If the firewall were to be enabled at this point the only access allowed would be the Guardian Digital WebTool from the administrator's host to the firewall and to machines on the internal network.

> **Note:** When defining a rule you are allowing the source to initiate a connection to an internal destination. *The firewall automatically allows a response from the internal destination to pass back to the source therefore you do not have to also create a rule that allows the destination to respond back to the source*. It is already done behind the scenes.

Figure 4-7. The Guardian Digital WebTool Rules



4.3.1.2. SSH

Now you know how to create an incoming rule. For our scenario we only want to allow incoming SSH access from the administrator's host to the internal network. This requires just one rule. Repeat the above procedure choosing the source host to be "*ADMIN*" and the destination hosts to be "*INTNET*". If you remember, we defined "*INTNET*" to be 10.0.99.0/24 - the internal network in the '*Hosts and Networks*" section (Section 4.2).

4.3.2. Outgoing Rules

Here are the specifications to create the rules that will allow the internal network to access the Internet. Just follow the rule creating procedures as above and plug in the Source, Destination and Protocol values as described for each service. For all of the following rules select the default source "*Port(s)*" of "*All Ports*".

4.3.2.1. SSH

Function: Let any host on the internal network SSH access to the Internet.

Table 4-1. SSH Outgoing Rule

Parameter	Value
Source	INTNET
Destination	EXTNET
Protocol	TCP

4.3.2.2. DNS

Function: Let only the name server submit DNS requests to the Internet.

Table 4-2. DNS Outgoing Rule

Parameter	Value
Source	NS

Parameter	Value
Destination	EXTNET
Protocol	TCP and UDP

Note: This will require two rules, one for the TCP protocol and another one for UDP.

4.3.2.3. SMTP

Function: Let only the mail server send mail to the Internet.

Note: For SMTP there is no existing choice in the destination "*Port(s)*" field so use the "*Specify*" field and enter "*25*" for the port number. Make sure you select the "*Specify*" radio button before saving this rule.

Table 4-3. SMTP Outgoing Rule

Parameter	Value
Source	MX
Destination	EXTNET
Protocol	TCP

4.3.2.4. FTP

Function: Let any internal host FTP to the Internet

Table 4-4. FTP Outgoing Rule

Parameter	Value
Source	INTNET
Destination	EXTNET
Protocol	TCP

Note: This will require two rules, one for the TCP protocol and another one for UDP.

4.3.2.5. HTTP

Function: Let any internal host connect to Internet web servers.

Table 4-5. HTTP Outgoing Rule

Parameter	Value
Source	INTNET
Destination	EXTNET

Parameter	Value
Protocol	TCP

4.3.2.6. HTTPS

Function: Let any internal host connect to Internet secure web servers.

Table 4-6. HTTPS Outgoing Rule

Parameter	Value
Source	INTNET
Destination	EXTNET
Protocol	TCP

4.3.2.7. NTP

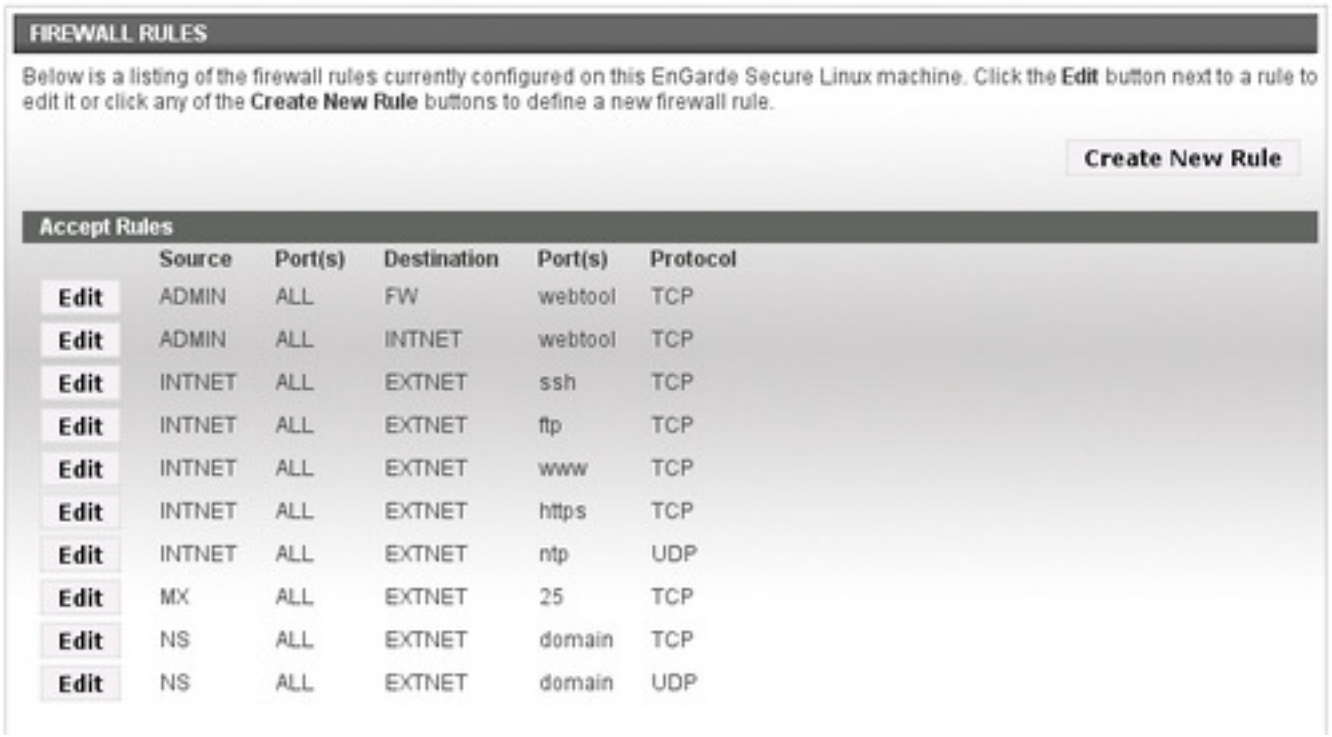
Function 1: Let any internal host connect to an Internet time server.

Table 4-7. NTP Outgoing Rule

Parameter	Value
Source	INTNET
Destination	EXTNET
Protocol	UDP

Note: Note NTP uses the UDP protocol.

Figure 4-8. Firewall Rules display



4.4. Port Forwarding

Port forwarding is needed when masquerading is used. What port forwarding does is to make the firewall look like a group of servers to the Internet while passing the packets to the appropriate servers on the internal private IP address space. For our example we will configure the following port forwarding.

> **Note:** When defining port forwarding for a service there is no need to also define any associated firewall rules to allow accept packets for the service. This is done automatically behind the scenes. This is why there were no rules created above to accommodate the following service, source, destination combinations.

- Port forward incoming SMTP requests to the mail server.
- Port forward incoming DNS requests to the name server.
- Port forward incoming SIMAP and SPOP3 requests to the mail server.
- Port forward incoming HTTP and HTTPS requests to the web server.
- Port forward incoming SMTP requests to a mail server.

4.4.1. Creating a Port Forwarding Rule

To create a port forwarding rule enter the "*Port Forwarding*" category in the Guardian Digital WebTool and click on "*Create Rule*". A pop-up window will appear titled "*Edit Port Forwarding Rule*". The procedure is pretty simple. Choose the following

parameters:

- Protocol that the service uses.
- The local address and port (local being the address and port that will receive requests for the service, in a two interface scenario this will be the external interface).
- The remote address and port to forward the packet to. In other words the IP address of the internal server and the associated port. Typically the local port and remote ports will be the same.

So for the first port forwarding rule we will forward SMTP requests that are sent from the Internet to the firewall. The rule will send these packets to the internal mail server which will then process the incoming mail. So fill in the following values in a new port forwarding rule pop-up window:

- Protocol for SMTP is TCP.
- Use the pull down menu for the local address and choose the external interface.
- The port used by SMTP is port 25 so enter 25 in the port field.
- The remote zone is internal so choose "int".
- The remote address is the address of the internal mail server and in our example that is 10.0.99.11.
- The remote port, which is the port that the internal mail server listens to for SMTP request is 25.
- Click on "Create Rule".

You will now see this rule listed under the "*Port Forwarding Rules*" title bar in the Guardian Digital WebTool page. So now any mail sent to the firewall from the Internet will automatically be forwarded to the internal mail server.

4.4.2. Creating the Remaining Port Forwarding Rules

Now create port forwarding rules for the rest of the services by repeating the above procedure and plugging in the following values fore each service.

4.4.2.1. DNS

Table 4-8. DNS Port Forwarding Spec

Parameter	Value
Protocol	TCP and UDP
Local Address	192.168.1.81
Local Port	53
Remote Zone	int
Remote Address	10.0.99.12
Remote Port	53

Note: DNS uses two protocols TCP and UDP so there needs to be two rules created, one for each protocol.

4.4.2.2. SIMAP

Table 4-9. SIMAP Port Forwarding Spec

Parameter	Value
Protocol	TCP
Local Address	192.168.1.81
Local Port	993
Remote Zone	int
Remote Address	10.0.99.11
Remote Port	993

4.4.2.3. SPOP3

Table 4-10. SPOP3 Port Forwarding Spec

Parameter	Value
Protocol	TCP
Local Address	192.168.1.81
Local Port	995
Remote Zone	int
Remote Address	10.0.99.11
Remote Port	995

4.4.2.4. HTTP

Table 4-11. HTTP Port Forwarding Spec

Parameter	Value
Protocol	TCP
Local Address	192.168.1.81
Local Port	80
Remote Zone	int
Remote Address	10.0.99.13
Remote Port	80

4.4.2.5. HTTPS

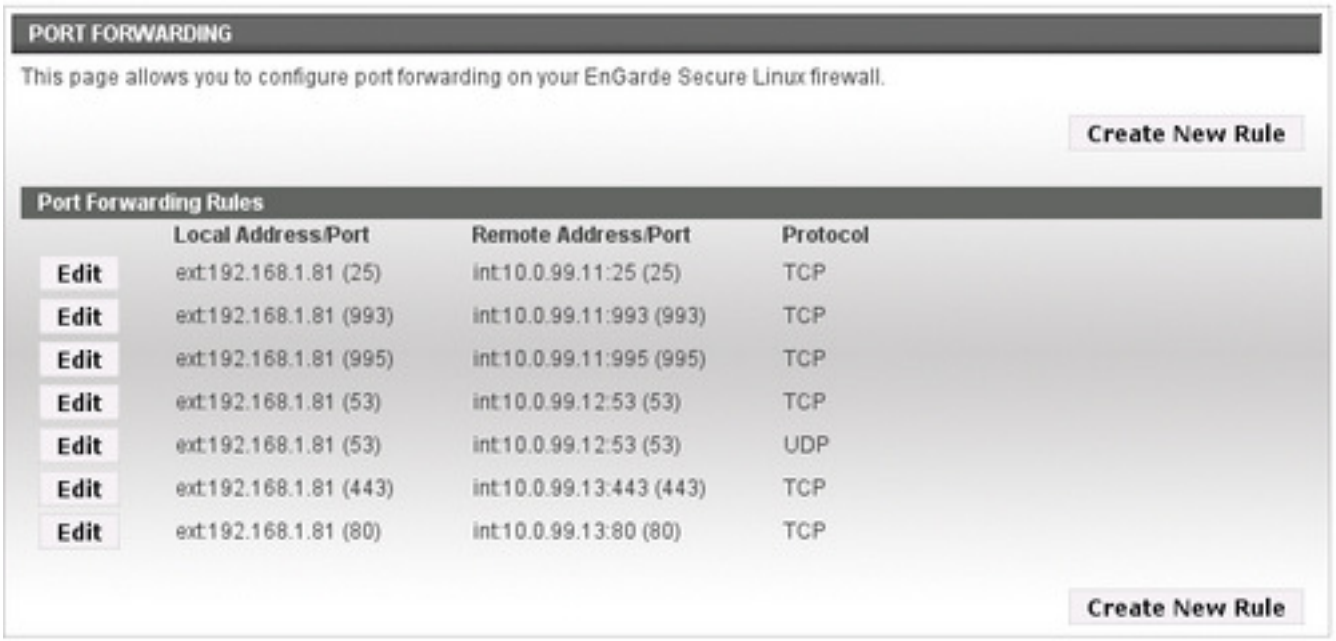
Table 4-12. HTTPS Port Forwarding Spec

Parameter	Value
Protocol	TCP
Local Address	192.168.1.81

Parameter	Value
Local Port	443
Remote Zone	int
Remote Address	10.0.99.13
Remote Port	443

Once these rules have been created you should now have a port forwarding table that looks like this.

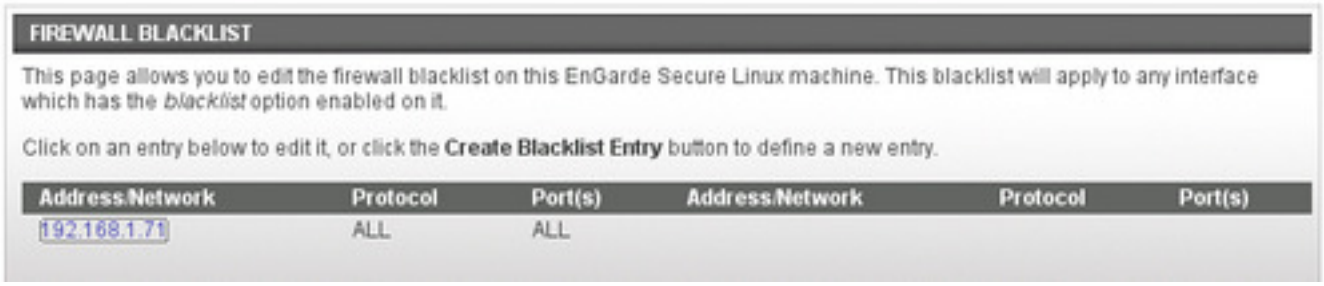
Figure 4-9. Port Forwarding Rules



4.5. Blacklisting

The function of blacklisting is to block some or all packets from a host or network. This is done in the "*Blacklist*" category. To create a blacklist simply click on "*Create a Blacklist Entry*" and a pop-up menu will appear. Enter the host IP or the network in CIDR notation. Choose between all protocols or specify either TCP or UDP. When specifying an protocol make sure the correct radio button is selected. The same goes for ports, you can choose all ports or specify a particular port. Click on "*Create Entry*" and you should now see the in the Blacklist WebTool page. In our example there is a malicious user on the host 192.168.1.71 in which I blocked all ports and all protocols sourced by this host.

Figure 4-10. Blacklist Rule



This procedure only defines an entity to be blacklisted. It by itself does not turn blacklisting on. To enable the blacklisting function you must go back to the "*General Configuration*" (Section 4.1) firewall category and check the blacklisting check box for the appropriate interface followed by clicking on "*Save Configuration*".

4.6. Starting and Stopping the Firewall

To start, stop and configure the firewall to automatically start on boot up go to the Guardian Digital WebTool main page and place the cursor over "*Services*" in the title bar. Click on "*Service Configuration*". To start the firewall manually click on the red text "*Stopped*" on the "*shorewall*" line in the "*Current State*" column. Once the firewall starts the red text will change to green text that says "*Running*". To stop the firewall click on the green "*Running*" and the firewall will be returned to the "*Stopped*" state. To enable the firewall to start on boot up click on the red "*Disabled*" in the "*Boot State*" column and the text will change to a green "*Enabled*". The firewall will now start on boot up. To disable this click on the "*Enabled*" and the text will turn to a red "*Disabled*" and the firewall will not automatically start on boot up.

Warning

When first experimenting with any new configuration you should seriously consider a couple of things. One, *until you have a configuration that is proven to let you into the firewall either by the Guardian Digital WebTool or SSH you may not want to have the firewall starting on boot up*. *It's no fun when you've accidentally locked your self out of your firewall.*

The second consideration is that you should have *console access* in the event that you do lock yourself out. *Configuring a firewall from a remote location is risky business*.

4.7. Testing the Firewall

Now that the firewall in configured start it up via the Guardian Digital WebTool. Now try to change pages in the Guardian Digital WebTool. If you can still use the Guardian Digital WebTool verify that the firewall is actually running. You can look at the running state in the "*Services Configuration*" WebTool page or if you have shell access as '*root*' you can run the command "*iptables -nL*". If you see output other than:

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
```

```
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

at least one rule is working. If all you see is the above then the firewall isn't running. You can also start the firewall manually as '*root*' by running the command "*/etc/init.d/shorewall start*". (To stop run "*/etc/init.d/shorewall stop*"). Once the firewall is running start testing by accessing the services that have been defined from both hosts that are allowed access and more importantly from hosts that shouldn't have access and verify proper operation. Also verify that any blacklisted hosts/networks don't have any access that has been denied. If all checks out you have just successfully configured an EnGarde Secure Linux firewall using the Guardian Digital WebTool.

Appendix A. Additional Resources

Here is a list of additional resources about shorewall and iptables configuration and use.

- Subscribe to the "enarde-users" maillist at http://infocenter.guardiandigital.com/community/ (http://infocenter.guardiandigital.com/community/)
- http://shorewall.viisage.com/
- http://www.shorewall.net/shorewall_quickstart_guide.htm
- http://www.iptables.org/
- http://www.netfilter.org/documentation/FAQ/netfilter-faq.html
- http://en.tldp.org/HOWTO/IP-Masquerade-HOWTO/