Chapter 21

Configuring Traffic Redirection on a Solaris Platform

This chapter describes how to redirect subscriber traffic by using redirect server on a Solaris platform. The chapter contains the following sections:

Installing the Redirect Server

To install and configure the redirect server on a Solaris platform:

1. Install the redirect server software as follows:
   - If you want to configure redundancy for the redirect server, install the software on two hosts.
   - If you do not want to configure redundancy for the redirect server, install the software on one host.

   For information about installing the Web redirect component of the captive portal system, see *SRC-PE Getting Started Guide, Chapter 27, Before You Install the SRC Software on a Solaris Platform*.
2. (Optional) Configure DNS.

Configuration Overview for Redirect Server

To configure the redirect server on a Solaris platform:

1. Configure IP Filter to define which traffic to redirect to the redirect server.

   See *Configuring IP Filter* on page 364.

2. If you plan to configure a redundant redirect server, make sure that you are familiar with the network configuration required.

   See *Chapter 19, Redirecting Subscriber Traffic*.

3. Configure how you want the redirect server to work in your environment:

   See *Configuring Redirect Server from the redir.properties File* on page 366.

Configuring IP Filter

If you run the SRC software on a Solaris platform, you use IP Filter to redirect subscriber requests for inappropriate or unsubscribed Web access. You specify Network Address Translation (NAT) rules in a configuration file that IP Filter uses to redirect traffic. When a packet arrives that matches a rule, its destination address is mapped as specified in the rule.

To install and configure IP Filter:

1. Install IP Filter on each server in which you want it to operate.

   For information about installing the IP Filter component of the captive portal system, see *SRC-PE Getting Started Guide, Chapter 28, Installing the SRC Software on a Solaris Platform*.

2. Access the IP Filter directory.

   cd /etc/opt/ipf

3. Create the NAT file */etc/opt/ipf /ipnat.conf*.

4. Add a rule to the *ipnat.conf* file to direct unauthorized traffic, and other rules, as needed to specify which traffic is to be redirected and to specify the destination for the redirected traffic.

   Create one rule for every interface on which redirected traffic can be received. For example, if you install the redirect server in a central location and set up multiple tunnel interfaces, you create one redirect rule for each tunnel interface. When you add rules to the *ipnat.conf* file, add a rule for authorized traffic followed by a rule for unauthorized traffic.

You can issue the **ifconfig -a** command to determine which network interfaces are configured on the host. You cannot use localhost (127.0.0.1) as a destination.

See the UNIX **man** pages for **ipnat** and **ipf** for more information about configuring IP Filter.

5. Update and start IP traffic filtering.

   /etc/init.d/ipfboot start

6. View active rules and sessions.

   /sbin/ipnat -l

The following sections give examples of the types of rules that you can configure by using IP Filter.

Example: Creating a Rule to Redirect Traffic to a Different Port Number

To enable subscribers to connect to the Web using the standard port, 80, for a Web server running on nonstandard port 8080, edit the *ipnat.conf* file on each Web server host to create a rule in the following format:

```
rdr ifName IpAddress/32 port 80 -> IpAddress port 8080 tcp
```

For example:

```
rdr hme0 192.168.1.1/32 port 80 > 192.168.1.1 port 8080 tcp
```

This rule filters legitimate traffic destined for the Web server and redirects it as follows:

1. Filters HTTP traffic that has a destination of standard port 80 and that meets the following criteria:

   - Has a destination of the specified masked IP address (the publicly known address for the Web server, stored in the JUNOSe router IP routing table).
   - Arrives on the primary network interface on the SAE host that receives traffic by means of the JUNOSe router.

2. Redirects filtered traffic to the specified target IP address on nonstandard port 8080.

   The target IP address must be an address that exists on the Web server and must be different from localhost (127.0.0.1).

Example: Creating a Rule to Redirect Unauthorized Traffic

To redirect invalid traffic, on each host in which you have installed the redirect server, add a rule to the *ipnat.conf* in the following format:

rdr *ifName* 0.0.0.0/0 port 80 -> *IpAddres*s port 8800 tcp

For example:

rdr hme0 0.0.0.0/0 port 80 > 10.227.1.163 port 8800 tcp

This rule redirects unauthorized traffic as follows:

1. Filters all HTTP packets that have the destination port of 80 and that meet the following criteria:
   - Has a destination of any IP address.
   - Arrives on the primary network interface on the redirect server host that receives traffic by means of the JUNOSe router.
2. Redirects packets to the specified target IP address on port 8800. The redirect server listens on this port and redirects subscribers to the captive portal page that you define to handle this traffic.

Configuring Redirect Server from the redir.properties File

If you run the SRC software on a Solaris platform, you configure the redirect server by editing the *redir.properties* file.

To configure the redirect server from the *redir.properties* file:

1. On each host on which you installed the redirect server software, access the directory in which you installed the redirect server, and run the configuration script.

   # cd /opt/UMC/redir
   # etc/config

2. Follow the instructions on the screen to configure the redirect server.

   Because the script includes some error checking, we recommend that you follow the instructions on the screen rather than directly editing the */opt/UMC/redir/etc/redir.properties* file.

   For information about the properties to be configured, see *Configuration Properties for the Redirect Server* on page 367.

   If you are configuring redundancy for the redirect server, assign one redirect server as the primary server, and the other as the redundant server.

   For information about getting information about the requests the redirect server is receiving and processing, see *Chapter 20, Configuring Traffic Redirection with the SRC CLI*.

Configuration Properties for the Redirect Server

You can modify the following properties for the redirect server from the configuration script that saves changes to the */etc/redir.properties* file.

redir.port

- TCP port on which the redirect server listens for requests.
- Value—Integer; valid port number in the range 1024–65535
- Default—8800

redir.url

- URL sent as a response to redirect requests. If redir.proxyurl is not configured, this URL is used for both proxied and nonproxied requests.
- Value—http://<serverHost>/accessDenied.do?url=%(url)
  - <serverHost>—Valid URL; string of ASCII characters.
- Guidelines—The URL can contain the special strings "%(url)s" and "%(proxy)s." If the HTTP request is sent to a proxy, the "%(url)s" string is replaced with the originally requested URL, and the "%(proxy)s" string is replaced with the proxy's "<ipAddress>:<port>". If the request is sent directly, the string is replaced with "None."
- Default—http://<serverHost>/accessDenied.do?url=%(url)

redir.proxy

- Configures proxy support. If you do not enable proxy support, the redirect server handles proxy requests in the same manner as direct requests.
- Value
  - Y—Proxy support is enabled.
  - N—Proxy support is disabled.
- Default—N

redir.proxyurl

- URL sent as a response to proxy requests. If you do not configure a value, then the URL defaults to the redir.url value. You can use this property to send proxy requests to a page different from the direct request page on the captive portal.
- Value—Valid URL; string of ASCII characters in URL string format
- Default—No value

redir.user

- Name of the user who owns the UNIX processes for the redirect server.
- Value—Text string
- Default—Nobody

redir.reqrate

- Number of requests that the redirect server can accept per minute from all clients (global sustained rate).
- Value—Integer in the range 0–2147483647
- Default—12000

redir.reqburst

- Maximum number of requests that the redirect server can accept from all clients (burst size). This value should exceed redir.reqrate. If the value for redir.reqrate exceeds this value, the redirect server drops the excess requests.
- Value—Integer in the range 0–2147483647
- Default—18000

redir.clientrate

- Number of requests that the redirect server can accept per minute for a single client (per client sustained rate).
- Value—Integer in the range 0–2147483647
- Default—25

redir.clientburst

- Maximum number of requests that the redirect server can accept for a single client (per client burst size). This value should exceed redir.clientrate.
- Value—Integer in the range 0–2147483647
- Default—50

redir.ext

- Specifies whether the redirect server should accept only URLs that point to files that have standard file extensions— < empty > , .asp, .htm, .html, .jsp, .php, .shtm, .shtml, and .xml. If you specify Y and the file does not have a standard file extension, the redirect server returns an HTTP 403 Forbidden message.
- Value
  - Y—Accepts only standard file extensions.
  - N—Accepts all file extensions.
- Default—N

redir.extensions

- List of additional file extensions. Employed only if you specified Y for redir.ext.
- Value—Text string consisting of acceptable file extensions separated by commas
- Default—No value

redir.monitor

- Configures redundancy for the redirect server.
- Value
  - Y—Enables redundancy
  - N—Specifies that only a single redirect server is used
- Default—N

monitor.host

- IP address or hostname for the redundant redirect server.
- Value—Fully qualified IP address or string
- Default—No value

monitor.virtualIp

- Configures virtual IP address of the redirect server. You must configure primary and redundant redirect servers to share this address under a common name in the DNS. Clients access the redirect server through this virtual IP address.
- Value—Fully qualified IP address
- Default—192.168.254.1

monitor.realIp

- Real IP address of the redirect server. When a primary redirect server is started, it dynamically establishes and maintains a static route on the client router to which it connects. The static route directs traffic destined for the virtual IP address of the server to the real IP address of the active redirect server.
- Value—Fully qualified IP address
- Default—Host IP address

monitor.master

- Specifies whether the redirect server identified in monitor.realIP is the primary redirect server.
- Value—Y or N
- Default—Y

monitor.checkInt

- Interval at which the redirect server polls the redundant redirect server.
- Value—Number of seconds in the range 60/<clientRate>–2147483647

  where <clientRate> is the number of requests per minute that the redirect engine accepts from one client
- Guidelines—Specifying a shorter time in the range leads to faster detection of problems and results in higher consumption of CPU resources.
- Default—30

ldap.url

- List of the URLs for directories employed by the redirect server.
- Value—Text string consisting of acceptable LDAP URLs in the format

  ldap://<host>:<portNumber>

  where <host> is the IP address or hostname of the directory host and <portNumber> is the TCP port
- Default—ldap://localhost

ldap.binddn

- Distinguished name (DN) that the redirect server uses to authorize connections to the directory.
- Value—Text string in LDAP format
- Default—*cn = ssp, ou = components, o = operators, o = umc*

ldap.bindpw

- Password that the redirect server uses to bind to the directory.
- Value—Text string
- Default—ssp

ldap.basedn

- Base DN that is the root of the directory tree.
- Value—Text string in LDAP format
- Default—*o = umc*

monitor.vrs

- Comma-separated list of virtual routers to which the redirect server connects.
- Value—Text string in the format

  <vrName>@<routerName>,<vrName>@<routerName>

  where <vrName> is the name of the virtual router and <routerName> is the name of the router on which the VR is configured
- Default—No value

dns.enable

- Controls the DNS server that is included with the redirect server.
- Value
  - Y—Starts the DNS server (only if proxy support is enabled)
  - N—Disables the DNS server
- Guidelines—Use this property only if you want to use the DNS server that is included with the redirect server. If you want to use another DNS server, do not enable the DNS server included with redirect server.
- Default—Y

dns.errorip

- IP address that is returned when a DNS request results in an unknown name (NXDOMAIN) error.
- Value—Fully qualified IP address
- Default—192.168.254.2

dns.forwarder

- DNS servers to which requests are forwarded.
- Value—Text string consisting of fully qualified IP addresses separated by commas
- Default—No value

dns.tcpport

- TCP port on which the DNS server listens.
- Value—Integer; valid port number in the range 1024—65535

  If you set the value to 0, no TCP socket is opened.
- Default—8853

dns.udpport

- UDP port on which the DNS server listens.
- Value—Integer; valid port number in the range 1024–65535

  If you set the value to 0, no UDP socket is opened.
- Default—8853

agent.path

- Path to the SNMP agent.
- Value— < directory path >
- Guidelines—If you install SRC components into the default directory structure, you do not need to change this value. You can change this value only by editing the */opt/UMC/redir.properties* file.
- Default—*.../agent/var*

redir.refresh

- Specifies whether the redirect server sends an HTTP 200 OK response or an HTML document that includes the < HTTP-Equiv = "Refresh" > header to a subscriber's browser in response to a captured request.
- Value
  - Y—Sends an HTTP 200 OK response with an HTML document that includes the < HTTP-Equiv = "Refresh" > header to a subscriber's browser in response to a captured request.
  - N— Sends an HTTP 302 Found response to a subscriber's browser in response to a captured request.
- Guidelines—By selecting Y, the load on the Web server is decreased because non-browser (or non-HTML) client applications that use HTTP do not follow this refresh message; however, most client applications do follow HTTP 302 messages.
- Default—Y

redir.refreshDoc

- Directory path to a local HTML file that the redirect server returns to a subscriber's browser in response to a captured request.
- Value— < path to HTML file >
- Guidelines—This property is used only if the redir.refresh property is set to Y.

  If you enter an invalid path, the redirect server uses a default file. This file can contain the string " %(url)s" which is replaced with the URL of the local HTML file to be returned to the subscriber's browser.
- Default—*etc/refresh.html*

Configuring Logging for Redirect Server

The redirect server logs incoming HTTP requests through the UNIX **syslog** command with a priority of INFO and log facility of LOCAL7. See *SRC-PE Monitoring and Troubleshooting Guide, Chapter 4, Configuring Logging for SRC Components on a Solaris Platform* for information about system logging.

Changing the Configuration for Redirect Server

If you change values set in the */opt/UMC/redir.properties* file, restart redirect server.

To restart redirect server:

1. Stop redirect server:

   /etc/rc2.d/S99UMCredirect stop

2. Start redirect server:

   /etc/rc2.d/S99UMCredirect start