Technical Tip

TT-0501401a -- Information -- 06-Jan-2005

NORTEL

Passive mode cookie-based persistence

Contents

Contents 1
Introduction 1
Associated Products 1
Overview 1
  Nortel Web and Application switch cookie persistency 3
  Passive cookie mode 3
Sample Configuration 4
  Setup 4
  Configuring the PC 4
  Configuring WebServer1 6
  Configuring WebServer2 7
  Configuring Alteon 8
    Configuring interfaces 9
    Configuring real servers 9
    Assigning real servers to a group 10
    Configuring the virtual server 10
    Enabling client/server processing on ports 11
    Enabling direct access mode 11
    Enabling SLB globally 11
  Testing the configuration 12
Appendix A. Alteon configuration file 21

Introduction

This document illustrates the operation and sample configuration of passive mode cookie-based persistence in Nortel Web and Application switches (formerly known as Alteon Web and Application switches).

Associated Products

The information in this document is intended to be used with the following product(s) with the indicated software or hardware revisions:

Product Name or Order Number	Revision Information	
	Potentially Affected	*Corrected*
Nortel Web and Application switches (formerly known as Alteon Web and Application switches): 180e, 180 Plus, 184, AD2, AD3, AD4, 2224, 2424, 2208, 2216, 3408	All	N/A

Overview

Cookies are a mechanism used to create a stateful session with HTTP requests and responses. Prior to the introduction of the cookie concept, HTTP servers would respond to each client request without relating that request to subsequent requests. The concept of cookies allows clients and servers that wish to exchange state information to place HTTP requests and responses within a larger context, or a so-called session. This context could be used to maintain a state of the session that could be necessary for some web applications. For example, a stateful session could be used for e-commerce, when the state would reflect the client's product selections or shopping cart. Or it could be used by web server administrators to count how many returning clients have visited the web site.

©2005 Nortel Networks Limited. All Rights Reserved

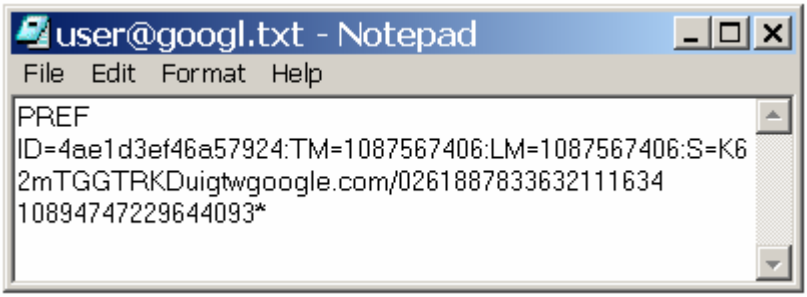
Cookies are strings that are placed into the HTTP header and passed from the server to the client's web browser. When the client initiates a connection to the server, it establishes a TCP session with a server and passes the HTTP request. A cookie-enabled server processes the HTTP request and sends the client an HTTP response containing an extra header: the Set-Cookie header. The Set-Cookie header contains:

- a cookie name and its value
- an optional comment
- an optional domain name (always starts with a leading dot)
- an optional max age of the cookie (if the max age is not specified in the Set-Cookie header, the user agent deletes the cookie as soon as the client closes the web browser)
- an optional path that specifies the subset of URLs which a particular cookie applies to
- an optional secure attribute that directs the user agent to use only a secure means to connect to the server
- a version attribute that identifies the version of the state management specification.

For example, below is a Set-Cookie header sent by the HTTP server at www.google.com:

```
Set-Cookie: PREF=ID=4ae1d3ef46a57924:TM=1087567406:LM=1087567406:S=K62mTGGTRKDuigtw;
expires=Sun, 17-Jan-2038 19:14:07 GMT; path=/; domain=.google.com\r\n
```

Once the client receives the cookie in the HTTP response, it stores it locally (unless configured to ignore cookies), usually in a form of a text file. Note that the location and form of storage depend on the web browser and operating system on a client's PC. For example, the received cookie from www.google.com could be stored as follows:



Once the cookie is received (assuming the client's web browser supports cookies), the client uses that cookie to send subsequent requests to the server. The cookie request header is used to send the cookie information back to the server. The cookie header contains the same attributes as the received Set-Cookie directive, with the exception of the comment, max age and secure attributes that are not sent back to the server.

For example, below is a subsequent request to the www.google.com:

```
Hypertext Transfer Protocol
    GET /images/logo.gif HTTP/1.1\r\n
    Accept: */*\r\n
    Referer: http://www.google.com/\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: www.google.com\r\n
    Connection: Keep-Alive\r\n
    Cookie:
PREF=ID=4ae1d3ef46a57924:TM=1087567406:LM=1087567406:S=K62mTGGTRKDuigtw\r\n \r\n
```

The server might ignore the received cookie or use it to determine the state of a session.

©2005 Nortel Networks Limited. All Rights Reserved

Nortel Web and Application switch cookie persistency

Nortel Web and Application switches support cookie-based persistency that allows the switch to redirect clients to the real server that initially processed the request. With cookie-based persistent load balancing, once an initial real server is selected based on the configured group metric, the switch forwards the subsequent requests based on the cookie header in the client's HTTP request.

The switch can operate in three modes of operation – insert cookie, passive cookie and rewrite cookie. This document illustrates the passive cookie mode. For information on insert or rewrite cookie modes, please consult the technical tips titled "Insert mode Cookie based persistence" or "Rewrite mode Cookie based persistence".

Passive cookie mode

When operating in passive cookie mode, the switch records the cookie value inserted by the real server and forwards the subsequent request from the client based on that cookie. The switch does not add or change the cookie, it just uses the cookie, thus operating in a passive mode.

When the client sends an initial HTTP request without any cookies in it (Figure 1), the switch makes a forwarding decision based on the configured load balancing metric for the real servers group.

The real server receives the request, processes it, inserts a cookie and sends an HTTP reply back to the client. The switch receives an HTTP reply, records the cookie value inserted by the real server and forwards the reply to the client PC. If the client needs to make a subsequent request to the server, it inserts the cookie request header into the HTTP request. When the switch receives an HTTP request containing the cookie value inserted previously by the real server, it looks up the real server associated with the cookie and forwards the request to the initially selected server. Thus persistency is achieved.

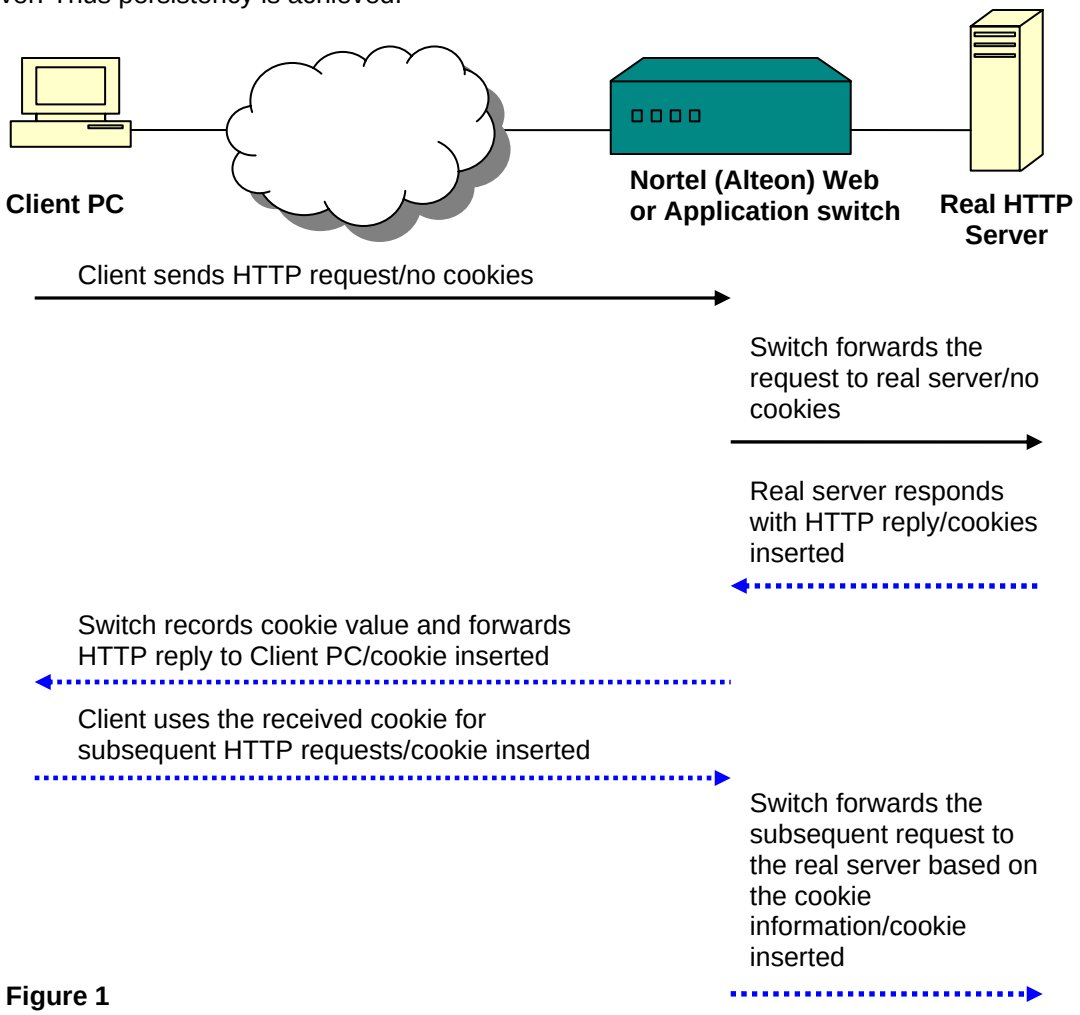
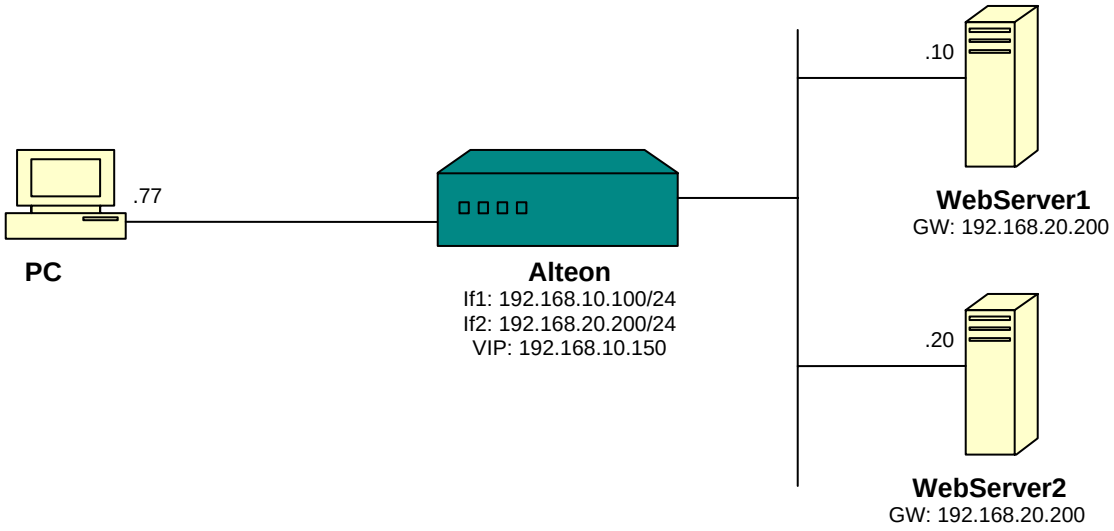


Figure 1

©2005 Nortel Networks Limited. All Rights Reserved

Sample Configuration

Setup



- **PC** – Windows 2000 Professional with Microsoft Internet Explorer 5.5 installed, IP address 192.168.10.77/24
- **Alteon** – Nortel (Alteon) Web switch AD3, code version 10.0.30.7, interface 1 IP 192.168.10.100/24, interface 2 IP 192.168.20.200/24, Virtual IP 192.168.10.150, port 1 – client port, port 8 – server port
- **WebServer1** – Linux RedHat 7.3 with Apache 1.3.31 software installed, IP address 192.168.20.10/24
- **WebServer2** – Linux RedHat 7.3 with Apache 1.3.31 software installed, IP address 192.168.20.20/24

The goal of the configuration is to configure passive mode cookie persistency on Alteon where the PC would be redirected to the same servers it was originally redirected to.

Configuring the PC

1. Configure the IP address (192.168.10.77/24) on the PC:

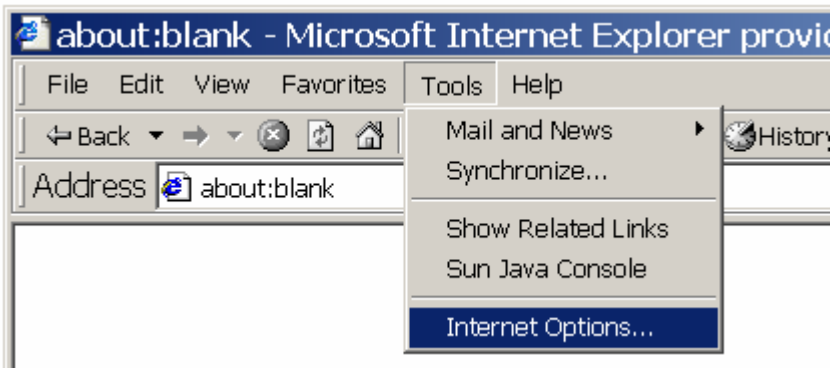
```
C:\>ipconfig

Windows 2000 IP Configuration
Ethernet adapter Local Area Connection:

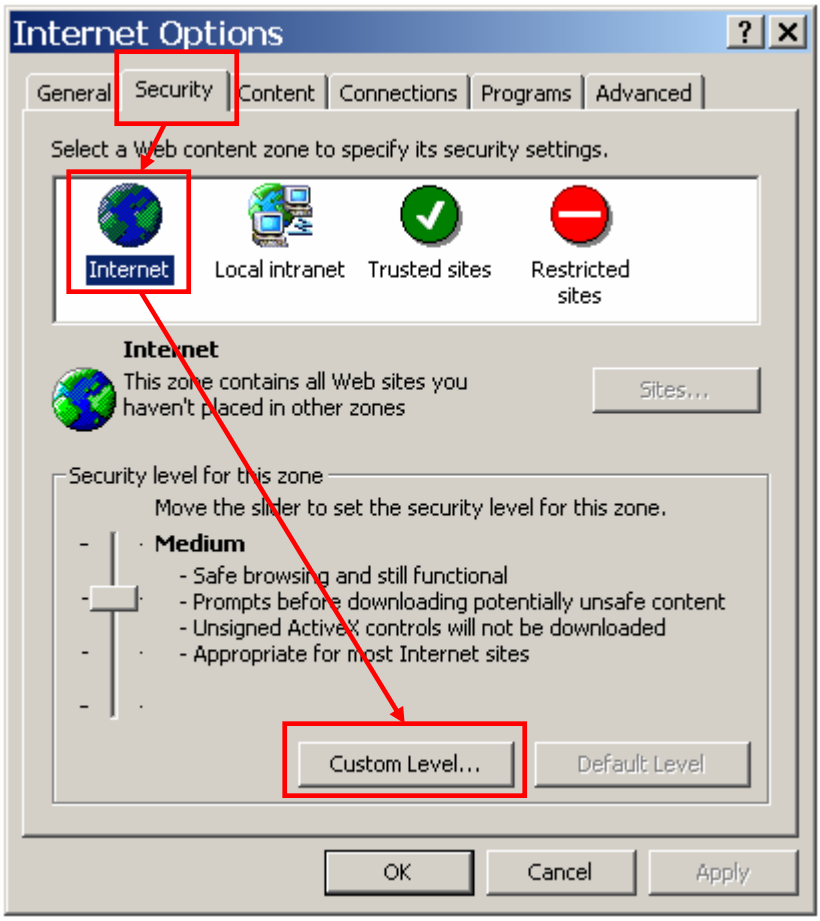
        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.10.77
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . :
```

©2005 Nortel Networks Limited. All Rights Reserved

2. It is assumed that cookies are enabled for the browser. If this is not the case, enable cookies:
   a) Navigate **Tools→Internet Options** from the browser menu:

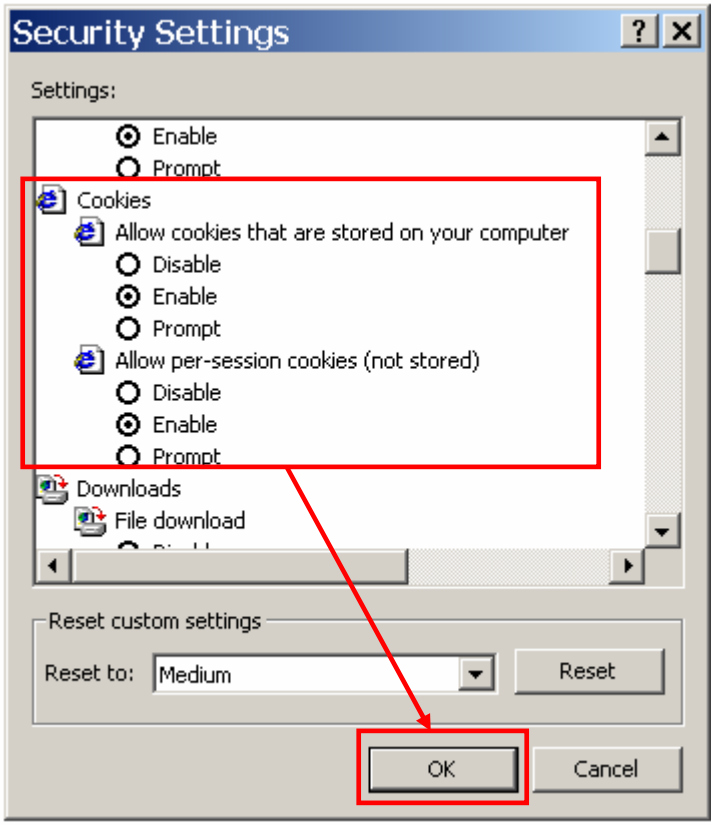


   b) The **Internet Options** window appears. Click on the **Security** tab.
   c) Select **Internet** and click **Custom Level**:



©2005 Nortel Networks Limited. All Rights Reserved

d) Scroll down to the **Cookies** section and select **Enable** for cookies.
e) Click **OK**:



f) In the **Internet Options** window, click **OK**.

Configuring WebServer1

1. Configure the IP address (192.168.20.10/24) on WebServer1:

```
[root@linux7 user]# /sbin/ifconfig eth0 192.168.20.10 netmask 255.255.255.0
```

2. Configure the Alteon interface 2 IP (192.168.20.200) as a default gateway:

```
[root@linux7 user]# /sbin/route add default gw 192.168.20.200
```

©2005 Nortel Networks Limited. All Rights Reserved

3. This configuration assumes that Apache has been compiled with the user tracking module enabled.
4. Enable user tracking. For that, add the user tracking directive to the httpd configuration file and turn the tracking on:

```
CookieTracking  on
```

```
[root@linux7 user]# tail /usr/sbin/conf/httpd.conf
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *:80>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    CustomLog logs/dummy-host.example.com-access log common
#</VirtualHost>

CookieTracking  on
```

Note: By default, when CookieTracking is enabled, Apache will add a cookie with the cookie name Apache and no expiration timer set.

5. Replace the default index.html file in Apache's root directory with a sample html file that will allow us to distinguish between the two web sites. For example:

```
<HTML>
 <HEAD>
  <TITLE>www.example-1.com</TITLE>
 </HEAD>
 <BODY>
  <H1 ALIGN="CENTER">www.example-1.com</H1>
 </BODY>
</HTML>
```

6. Start the HTTP server (assuming Apache has been installed with /usr/sbin/ prefix):

```
[root@linux7 sbin]# /usr/sbin/bin/apachectl start
```

Configuring WebServer2

1. Configure the IP address (192.168.20.20/24) on WebServer2:

```
[root@linux7 user]# /sbin/ifconfig eth0 192.168.20.20 netmask 255.255.255.0
```

2. Configure Alteon interface 2 IP (192.168.20.200) as a default gateway:

```
[root@linux7 user]# /sbin/route add default gw 192.168.20.200
```

©2005 Nortel Networks Limited. All Rights Reserved

3. This configuration assumes that Apache has been compiled with the user tracking module enabled.
4. Enable user tracking. For that, add the user tracking directive to the httpd configuration file and turn the tracking on:

```
CookieTracking  on
```

```
[root@linux7 user]# tail /usr/sbin/conf/httpd.conf
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *:80>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    CustomLog logs/dummy-host.example.com-access log common
#</VirtualHost>

CookieTracking  on
```

Note: By default, when CookieTracking is enabled, Apache will add a cookie with the cookie name “Apache” and no expiration timer set. The default behavior is used in this example. If there is a need to set a different name for the cookie, use the CookieName directive. If there is a need to set the expiration timer, use the CookieExpires directive. For more information on cookies with Apache, please consult the Apache web server documentation available at httpd.apache.org.

5. Replace the default index.html file in the root directory with a sample html file that will allow us to distinguish between the two web sites. For example:

```
<HTML>
 <HEAD>
  <TITLE>www.example-2.com</TITLE>
 </HEAD>
 <BODY>
  <H1 ALIGN="CENTER">www.example-2.com</H1>
 </BODY>
</HTML>
```

6. Start the HTTP server (assuming Apache has been installed with /usr/sbin prefix):

```
[root@linux7 sbin]# /usr/sbin/bin/apachectl start
```

Configuring Alteon

Note: Configuration of Alteon has been started from factory default settings.

1. Log in to the switch:

```
Enter password:
System Information at 11:05:11 Wed Apr 21, 2004
Alteon AD3
sysName:
sysLocation:
. . .
```

©2005 Nortel Networks Limited. All Rights Reserved

2. Select **n** to not run the setup script:

```
The switch is booted with factory default configuration.
  To ease the configuration of the switch, a "Set Up" facility which
  will prompt you with those configuration items that are essential
  to the operation of the switch is provided.
Would you like to run "Set Up" to configure the switch? [y/n] n
```

Configuring interfaces

1. Configure the IP address (192.168.10.100/24) for interface 1 and enable the interface:

```
>> Main# /cfg/ip/if 1/mask 255.255.255.0/addr 192.168.10.100/ena
Current subnet mask:     0.0.0.0
New pending subnet mask: 255.255.255.0
Current IP address:     0.0.0.0
New pending IP address: 192.168.10.100
Pending new broadcast address:  192.168.10.255
Current status: disabled
New status:     enabled
```

2. Configure the IP address (192.168.20.200/24) for interface 2 and enable the interface:

```
>> IP Interface 1# /cfg/ip/if 2/mask 255.255.255.0/addr 192.168.20.200/ena
Current subnet mask:     0.0.0.0
New pending subnet mask: 255.255.255.0
Current IP address:     0.0.0.0
New pending IP address: 192.168.20.200
Pending new broadcast address:  192.168.20.255
Current status: disabled
New status:     enabled
```

3. Apply and save the changes:

```
>> IP Interface 2# apply/save
------------------------------------------------------------------
Apply complete; don't forget to "save" updated configuration.
Request will first copy the FLASH "active" config to "backup",
 then overlay FLASH "active" with new config.
Confirm saving to FLASH [y/n]: y
New config successfully saved to FLASH.
Switch is currently set to use factory default config block on next boot.
Do you want to change that to the active config block? [y/n] y
Next boot will use active config block.
```

Configuring real servers

1. Configure real server 1. Enter the real server 1 configuration menu, set the real server 1 IP address (192.168.20.10) and enable the real server:

```
>> IP Interface 2# /cfg/slb/real 1/rip 192.168.20.10/ena
Current real server IP address:     0.0.0.0
New pending real server IP address: 192.168.20.10
Current status: disabled
New status:     enabled
```

2. Configure real server 2. Enter the real server 2 configuration menu, set the real server 2 IP address (192.168.20.20) and enable the real server:

```
>> Real server 1 # /cfg/slb/real 2/rip 192.168.20.20/ena
Current real server IP address:     0.0.0.0
New pending real server IP address: 192.168.20.20
Current status: disabled
New status:     enabled
```

©2005 Nortel Networks Limited. All Rights Reserved

Assigning real servers to a group

1. Define a real servers group, and place real server 1 and 2 into this group:

```
>> Real server 2 # /cfg/slb/group 1/add 1/add 2
Real server 1 added to real server group 1.
Real server 2 added to real server group 1.
```

2. Apply the appropriate metric for load balancing. If the IP address of the real server handling the request is to be embedded into a cookie, select round robin or least connections metric type. If the IP address is not to be embedded, select hash metric. In this example, round robin metric is used:

```
>> Real server group 1# metric round
Current metric:         leastconns
New pending metric:     roundrobin
```

3. Set the health check to TCP:

```
>> Real server group 1# health tcp
Current health check type:      tcp
New pending health check type:  tcp
```

Configuring the virtual server

1. Configure the virtual server (server 1 in this example). Set the IP address for the virtual server, and enable the virtual server:

```
>> Real server group 1# /cfg/slb/virt 1/vip 192.168.10.150/ena
Current virtual server IP address:     0.0.0.0
New pending virtual server IP address: 192.168.10.150
Current status: disabled
New status:     enabled
```

2. Assign HTTP service to the virtual server and assign the created group 1 to the server:

```
>> Virtual Server 1# service http/group 1
Current real server group:     1
New pending real server group: 1
```

3. Enable delayed binding. Delayed binding is required as layer 7 information (HTTP) is not sent in the first packet from the client:

```
>> Virtual Server 1 http Service# dbind ena
Current delayed binding: disabled
New delayed binding:     enabled
```

©2005 Nortel Networks Limited. All Rights Reserved

4. Enable persistent binding using the **/cfg/slb/virt *<virtual server number>* /service *<service>*/pbind** command.
    a) When prompted, enter **cookie** as a persistence mode.
    b) Enter **p** to select passive mode of operation.
    c) Enter the cookie name to look for in the real servers reply. As we didn't change the default cookie name for the Apache real server, **Apache** is used as the name.
    d) Enter the starting point of the cookie value to use for hashing (1 in this example).
    e) Enter the number of bytes to extract (32 in this example; it will select the whole cookie value). This specifies how many bytes (starting from the starting point specified above) should be used by Alteon for hashing to determine a real server.
    f) **Disable** the **Look for cookie in URI parameter**, as there is no need to enable a cookie search in URI as the cookie will not be embedded in the URL for this configuration:

```
>> Virtual Server 1 http Service# /cfg/slb/virt 1/service 80/pbind
Current persistent binding mode: disabled
Enter clientip|cookie|sslid|disable persistence mode:   cookie
Enter passive|rewrite|insert cookie persistence mode [p/r/i]:   p
Enter Cookie Name:      Apache
Enter the starting point of the cookie value [1-64]:    1
Enter number of bytes to extract [1-64]: 32
Look for cookie in URI [e|d]:   d
```

Enabling client/server processing on ports

Enable client/server processing on the appropriate ports. In this configuration, the client is connected to port 1 and both servers are connected through the hub to port 8. Thus client processing should be enabled on port 1 and server processing should be enabled on port 8:

```
>> Virtual Server 1 http Service# /cfg/slb/port 1/client e
Current client processing: disabled
New client processing:     enabled

>> SLB port 7# /cfg/slb/port 8/server e
Current server processing: disabled
New server processing:     enabled
```

Enabling direct access mode

Enable direct access mode (DAM), required for the delayed binding to be enabled:

```
>> SLB port 1# /cfg/slb/adv/direct e
Current Direct Access Mode: disabled
New Direct Access Mode:     enabled
```

Note: Alternatively, Proxy IP addresses can be assigned to every port and direct access mode can be disabled.

Enabling SLB globally

1. Enable SLB globally:

```
>> Layer 4 Advanced# /cfg/slb/on
Current status: OFF
New status:     ON
```

©2005 Nortel Networks Limited. All Rights Reserved

2. Apply and save the changes:

```
>> Layer 4# apply/save
------------------------------------------------------------------
Apply complete; don't forget to "save" updated configuration.
Request will first copy the FLASH "active" config to "backup",
 then overlay FLASH "active" with new config.
Confirm saving to FLASH [y/n]:y
New config successfully saved to FLASH.
```

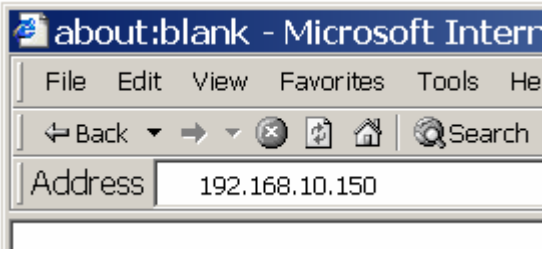
Note: As soon as the changes are applied, the following messages appear on the console indicating that the real servers and virtual server are up and operational:

```
Jun 21 10:59:04 NOTICE  slb: real service 192.168.20.10:80 operational
Jun 21 10:59:04 NOTICE  slb: real server 192.168.20.10 operational
Jun 21 10:59:04 NOTICE  slb: Services are available for Virtual Server
1:192.168.10.150
Jun 21 10:59:05 NOTICE  slb: real service 192.168.20.20:80 operational
Jun 21 10:59:05 NOTICE  slb: real server 192.168.20.20 operational
```

Testing the configuration

1. Start traffic capture on the PC using, for example, Ethereal. Start traffic capture on the WebServers using, for example, tcpdump.

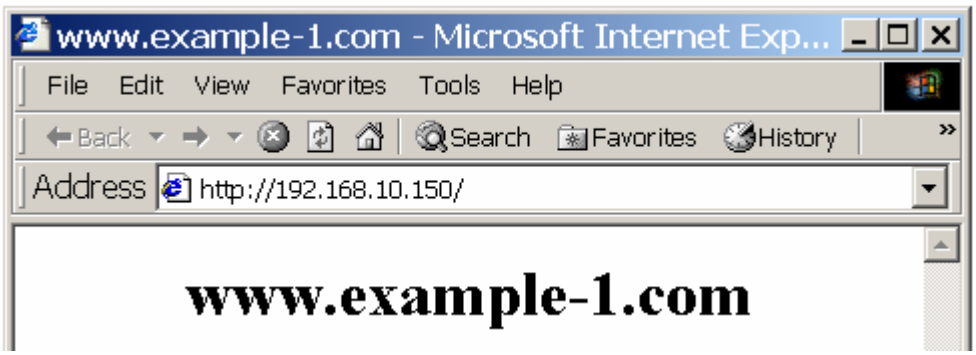
2. Open a web browser and enter the IP address of the virtual server (192.168.10.150):



3. While the page loads, check the session table on Alteon., Alteon has selected WebServer1 (192.168.20.10) to process the request:

```
>> Session Table Information# /info/slb/sess/dump
 8,4:  192.168.10.77 3040, 192.168.10.150 80 -> 4100 192.168.20.10 80  age 10
```

4. The WebServer1 page loads:



5. Refresh the page and check the session table again:

```
>> Session Table Information# /info/slb/sess/dump
```

©2005 Nortel Networks Limited. All Rights Reserved

```
 8,4:  192.168.10.77 3041, 192.168.10.150 80 -> 4100 192.168.20.10 80  age 10
```

6. Let's take a look at the captured traffic:

The PC (192.168.10.77) initiates an HTTP session by sending a TCP SYN packet to the virtual server at 192.168.10.150:

```
Frame 119 (62 bytes on wire, 62 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:60:cf:46:53:6e
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.10.150
(192.168.10.150)
Transmission Control Protocol, Src Port: 3040 (3040), Dst Port: http (80), Seq:
4033201338, Ack: 0, Len: 0
    Source port: 3040 (3040)
    Destination port: http (80)
    Sequence number: 4033201338
    Header length: 28 bytes
    Flags: 0x0002 (SYN)
    Window size: 65535
    Checksum: 0x2b9b (correct)
    Options: (8 bytes)
```

Due to enabled delayed binding, Alteon proceeds with a handshake on behalf of the real server and responds to the PC with a TCP ACK:

```
Frame 120 (62 bytes on wire, 62 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:6e, Dst: 00:08:74:9a:e5:85
Internet Protocol, Src Addr: 192.168.10.150 (192.168.10.150), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 3040 (3040), Seq:
12682932, Ack: 4033201339, Len: 0
    Source port: http (80)
    Destination port: 3040 (3040)
    Sequence number: 12682932
    Acknowledgement number: 4033201339
    Header length: 24 bytes
    Flags: 0x0012 (SYN, ACK)
    Window size: 65535
    Checksum: 0xb91b (correct)
    Options: (4 bytes)
```

The client acknowledges the SYN ACK with an ACK:

```
Frame 121 (54 bytes on wire, 54 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:60:cf:46:53:6e
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.10.150
(192.168.10.150)
Transmission Control Protocol, Src Port: 3040 (3040), Dst Port: http (80), Seq:
4033201339, Ack: 12682933, Len: 0
    Source port: 3040 (3040)
    Destination port: http (80)
    Sequence number: 4033201339
    Acknowledgement number: 12682933
    Header length: 20 bytes
    Flags: 0x0010 (ACK)
    Window size: 65535
    Checksum: 0xd0d8 (correct)
```

©2005 Nortel Networks Limited. All Rights Reserved

Once a handshake is complete, the client PC sends an HTTP request:

```
Frame 122 (433 bytes on wire, 433 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:60:cf:46:53:6e
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.10.150
(192.168.10.150)
Transmission Control Protocol, Src Port: 3040 (3040), Dst Port: http (80), Seq:
4033201339, Ack: 12682933, Len: 379
Hypertext Transfer Protocol
    GET / HTTP/1.1\r\n
    Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-
excel, application/vnd.ms-powerpoint, application/msword, application/x-shockwave-
flash, */*\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: 192.168.10.150\r\n
    Connection: Keep-Alive\r\n
    \r\n
```

Once an HTTP request is received by the Alteon, it checks whether the request already contains a cookie header with real server information. As this is an initial request, no cookie information could be obtained and Alteon selects a server for load balancing based on the configured for the group metric. In this example, WebServer1 (192.168.20.10) was selected.

Once the real server is selected, Alteon starts the TCP handshake with the real server on behalf of the client by sending a TCP SYN:

```
Frame 123 (64 bytes on wire, 64 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:0c:29:4d:f9:8c
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.20.10
(192.168.20.10)
Transmission Control Protocol, Src Port: 4100 (4100), Dst Port: http (80), Seq:
4033201338, Ack: 0, Len: 0
    Source port: 4100 (4100)
    Destination port: http (80)
    Sequence number: 4033201338
    Header length: 24 bytes
    Flags: 0x0002 (SYN)
    Window size: 65535
    Checksum: 0x330a (correct)
    Options: (4 bytes)
```

©2005 Nortel Networks Limited. All Rights Reserved

The real server (192.168.20.10) responds with a SYN ACK:

```
Frame 124 (58 bytes on wire, 58 bytes captured)
Ethernet II, Src: 00:0c:29:4d:f9:8c, Dst: 00:60:cf:46:53:60
Internet Protocol, Src Addr: 192.168.20.10 (192.168.20.10), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 4100 (4100), Seq:
1144429854, Ack: 4033201339, Len: 0
    Source port: http (80)
    Destination port: 4100 (4100)
    Sequence number: 1144429854
    Acknowledgement number: 4033201339
    Header length: 24 bytes
    Flags: 0x0012 (SYN, ACK)
    Window size: 5840
    Checksum: 0x3ad4 (correct)
    Options: (4 bytes)
```

Alteon acknowledges it with an ACK:

```
Frame 125 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:0c:29:4d:f9:8c
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.20.10
(192.168.20.10)
Transmission Control Protocol, Src Port: 4100 (4100), Dst Port: http (80), Seq:
4033201339, Ack: 1144429855, Len: 0
    Source port: 4100 (4100)
    Destination port: http (80)
    Sequence number: 4033201339
    Acknowledgement number: 1144429855
    Header length: 24 bytes
    Flags: 0x0010 (ACK)
    Window size: 65535
    Checksum: 0x51a5 (correct)
    Options: (4 bytes)
```

Once the TCP handshake is complete, Alteon sends the client's HTTP request to the real server:

```
Frame 126 (433 bytes on wire, 433 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:0c:29:4d:f9:8c
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.20.10
(192.168.20.10)
Transmission Control Protocol, Src Port: 4100 (4100), Dst Port: http (80), Seq:
4033201339, Ack: 1144429855, Len: 379
Hypertext Transfer Protocol
    GET / HTTP/1.1\r\n
    Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-
excel, application/vnd.ms-powerpoint, application/msword, application/x-shockwave-
flash, */*\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: 192.168.10.150\r\n
    Connection: Keep-Alive\r\n
    \r\n
```

©2005 Nortel Networks Limited. All Rights Reserved

The real server acknowledges the request, processes the request, inserts a cookie into the HTTP response and replies to the client:

```
Frame 129 (707 bytes on wire, 707 bytes captured)
Ethernet II, Src: 00:0c:29:4d:f9:8c, Dst: 00:60:cf:46:53:60
Internet Protocol, Src Addr: 192.168.20.10 (192.168.20.10), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 4100 (4100), Seq:
1144429855, Ack: 4033201718, Len: 653
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 15:44:20 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n
    Set-Cookie: Apache=192.168.10.77.104251087832660509; path=/\r\n
    Last-Modified: Mon, 21 Jun 2004 15:15:00 GMT\r\n
    ETag: "196a2-7a-40d6fb74;40d6fb74"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 122\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html
```

Alteon receives an HTTP reply on its server-enabled port, looks up the cookie with the name of Apache (specified during configuration), records the cookie value and associated real server and forwards the request to the client PC:

```
Frame 130 (707 bytes on wire, 707 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:08:74:9a:e5:85
Internet Protocol, Src Addr: 192.168.10.150 (192.168.10.150), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 3040 (3040), Seq:
12682933, Ack: 4033201718, Len: 653
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 15:44:20 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n
    Set-Cookie: Apache=192.168.10.77.104251087832660509; path=/\r\n
    Last-Modified: Mon, 21 Jun 2004 15:15:00 GMT\r\n
    ETag: "196a2-7a-40d6fb74;40d6fb74"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 122\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html
```

©2005 Nortel Networks Limited. All Rights Reserved

The client receives the response with a cookie, and as it is configured to allow per-session cookies, it inserts that cookie into subsequent HTTP requests when the screen is refreshed:

```
Frame 635 (346 bytes on wire, 346 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:60:cf:46:53:6e
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.10.150
(192.168.10.150)
Transmission Control Protocol, Src Port: 3041 (3041), Dst Port: http (80), Seq:
4048564599, Ack: 2831237901, Len: 292
Hypertext Transfer Protocol
    GET / HTTP/1.1\r\n
    Accept: */*\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: 192.168.10.150\r\n
    Connection: Keep-Alive\r\n
    Cache-Control: no-cache\r\n
    Cookie: Apache=192.168.10.77.104251087832660509\r\n
    \r\n
```

Alteon sees the cookie it has issued earlier and forwards the request to the same real server:

```
Frame 639 (346 bytes on wire, 346 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:0c:29:4d:f9:8c
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.20.10
(192.168.20.10)
Transmission Control Protocol, Src Port: 4100 (4100), Dst Port: http (80), Seq:
4048564599, Ack: 1205845126, Len: 292
Hypertext Transfer Protocol
    GET / HTTP/1.1\r\n
    Accept: */*\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: 192.168.10.150\r\n
    Connection: Keep-Alive\r\n
    Cache-Control: no-cache\r\n
    Cookie: Apache=192.168.10.77.104251087832660509\r\n
    \r\n
```

©2005 Nortel Networks Limited. All Rights Reserved

The real server responds with an HTTP reply:

```
Frame 642 (585 bytes on wire, 585 bytes captured)
Ethernet II, Src: 00:0c:29:4d:f9:8c, Dst: 00:60:cf:46:53:60
Internet Protocol, Src Addr: 192.168.20.10 (192.168.20.10), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 4100 (4100), Seq:
1205845126, Ack: 4048564891, Len: 531
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 15:45:21 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n
    Last-Modified: Mon, 21 Jun 2004 15:15:00 GMT\r\n
    ETag: "196a2-7a-40d6fb74;40d6fb74"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 122\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html

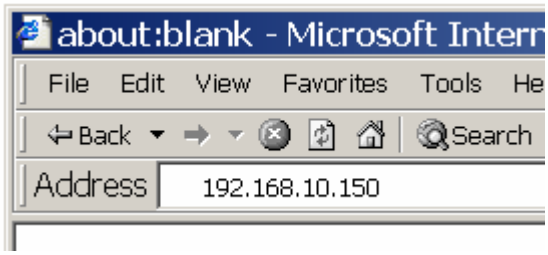
Frame 643 (585 bytes on wire, 585 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:08:74:9a:e5:85
Internet Protocol, Src Addr: 192.168.10.150 (192.168.10.150), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 3041 (3041), Seq:
2831237901, Ack: 4048564891, Len: 531
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 15:45:21 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n
    Last-Modified: Mon, 21 Jun 2004 15:15:00 GMT\r\n
    ETag: "196a2-7a-40d6fb74;40d6fb74"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 122\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html
```

7. Close the web browser. (As we did not configure any expiration time for the cookies on the Apache servers, the default “None” expiration time is applied. Therefore, the cookie expires with a web browser closure.)

8. Start the traffic capture.

©2005 Nortel Networks Limited. All Rights Reserved

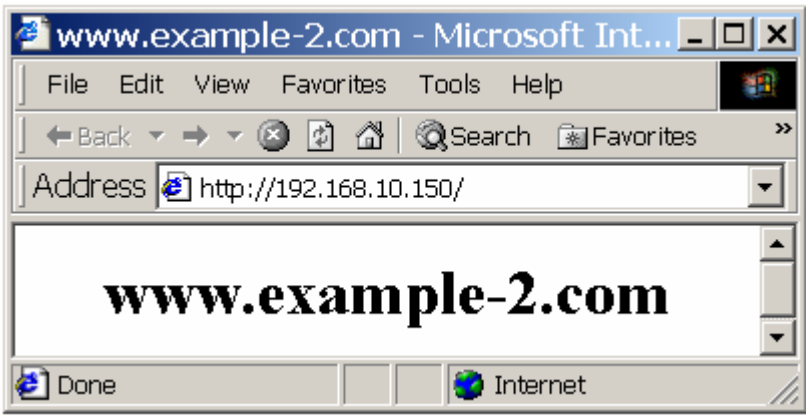
9. Open a new browser window and initiate another connection to the virtual server 192.168.10.150:



10. While the page loads, check the session table on Alteon. Since round robin has been configured as the load balancing algorithm and since the cookie has expired, the subsequent client request has been forwarded to the next available real server – WebServer2 (192.168.20.20):

```
>> Session Table Information# /info/slb/sess/dump
 8,6:  192.168.10.77 3043, 192.168.10.150 80 -> 4102 192.168.20.20 80  age 10
```

11. The WebServer2 page loads:



12. Take a look at the previous capture. As with the previous example, at first, Alteon establishes a TCP connection with the client. Once the TCP handshake is established, the client sends an HTTP request. Note that no cookie information is present in the request, as the cookie has expired with the closure of the browser:

```
Frame 1073 (433 bytes on wire, 433 bytes captured)
Ethernet II, Src: 00:08:74:9a:e5:85, Dst: 00:60:cf:46:53:6e
Internet Protocol, Src Addr: 192.168.10.77 (192.168.10.77), Dst Addr: 192.168.10.150
(192.168.10.150)
Transmission Control Protocol, Src Port: 3043 (3043), Dst Port: http (80), Seq:
4061832229, Ack: 1377930680, Len: 379
Hypertext Transfer Protocol
    GET / HTTP/1.1\r\n
    Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-
excel, application/vnd.ms-powerpoint, application/msword, application/x-shockwave-
flash, */*\r\n
    Accept-Language: en-us\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461; PC SRP
2.0)\r\n
    Host: 192.168.10.150\r\n
    Connection: Keep-Alive\r\n
    \r\n
```

©2005 Nortel Networks Limited. All Rights Reserved

Upon receiving an HTTP request and with the absence of cookies, Alteon decides which real server to use and splits the session to the selected real server. The real server processes the request and replies with an HTTP reply that has a cookie inserted:

```
Frame 1080 (703 bytes on wire, 703 bytes captured)
Ethernet II, Src: 00:0c:29:63:8f:59, Dst: 00:60:cf:46:53:60
Internet Protocol, Src Addr: 192.168.20.20 (192.168.20.20), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 4102 (4102), Seq:
3095198925, Ack: 4061832608, Len: 649
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 01:34:07 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n

    Set-Cookie: Apache=192.168.10.77.93891087781647741; path=/\r\n
    Last-Modified: Mon, 21 Jun 2004 01:08:38 GMT\r\n
    ETag: "196a0-78-40d63516;40d63516"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 120\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html
```

Alteon receives a reply on its server ports, records the cookie value and the server, and forwards the reply to the client:

```
Frame 1081 (703 bytes on wire, 703 bytes captured)
Ethernet II, Src: 00:60:cf:46:53:60, Dst: 00:08:74:9a:e5:85
Internet Protocol, Src Addr: 192.168.10.150 (192.168.10.150), Dst Addr: 192.168.10.77
(192.168.10.77)
Transmission Control Protocol, Src Port: http (80), Dst Port: 3043 (3043), Seq:
1377930680, Ack: 4061832608, Len: 649
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 21 Jun 2004 01:34:07 GMT\r\n
    Server: Apache/1.3.31 (Unix)\r\n
    Content-Location: index.html.en\r\n
    Vary: negotiate,accept-language,accept-charset\r\n
    TCN: choice\r\n
    Set-Cookie: Apache=192.168.10.77.93891087781647741; path=/\r\n
    Last-Modified: Mon, 21 Jun 2004 01:08:38 GMT\r\n
    ETag: "196a0-78-40d63516;40d63516"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 120\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n
    Content-Language: en\r\n
    \r\n
Line-based text data: text/html
```

©2005 Nortel Networks Limited. All Rights Reserved

Appendix A. Alteon configuration file

```
script start "Alteon AD3" 4  /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 14:15:26 Mon Jun 21, 2004
/* Version 10.0.30.7,  Base MAC address 00:60:cf:46:53:60
/c/sys
        idle 60
/c/ip/if 1
        ena
        addr 192.168.10.100
/c/ip/if 2
        ena
        addr 192.168.20.200
/c/slb
        on
/c/slb/adv
        direct ena
/c/slb/real 1
        ena
        rip 192.168.20.10
/c/slb/real 2
        ena
        rip 192.168.20.20
/c/slb/group 1
        metric roundrobin
        add 1
        add 2
/c/slb/port 1
        client ena
/c/slb/port 8
        server ena
/c/slb/virt 1
        ena
        vip 192.168.10.150
/c/slb/virt 1/service http
        group 1
        dbind ena
/c/slb/virt 1/service 80/pbind cookie passive Apache 1 32 disable
/c/slb/virt 1/service 80/rcount 1
/
script end  /**** DO NOT EDIT THIS LINE!
```

©2005 Nortel Networks Limited. All Rights Reserved

Copyright © 2005 Nortel Networks Limited - All Rights Reserved.

Nortel, Nortel Networks, the Nortel logo, Globemark, and Alteon are trademarks of Nortel Networks Limited.

Microsoft, Microsoft Internet Explorer and Windows are trademarks of Microsoft Corporation.

RedHat is a trademark of Red Hat, Inc.

Apache is a trademark of The Apache Software Foundation.

The information in this document is subject to change without notice. The statements, configurations, technical data, and recommendations in this document are believed to be accurate and reliable, but are presented without express or implied warranty. Users must take full responsibility for their applications of any products specified in this document. The information in this document is proprietary to Nortel Networks Limited.

To access more technical documentation, search our knowledge base, or open a service request online, please visit Nortel Networks Technical Support on the web at: http://www.nortel.com/support

If after following this guide you are still having problems, please ensure you have carried out the steps exactly as in this document. If problems still persist, please contact Nortel Networks Technical Support (contact information is available online at http://www.nortel.com/cgi-bin/comments/comments.cgi?key=techsupport_cu).

We welcome you comments and suggestions on the quality and usefulness of this document. If you would like to leave a feedback please send your comments to: CRALT@nortel.com