radware

Alteon
34.x

# Alteon Level 1
# Lab Manual
# Server Load Balancing

# Table of Contents

# Objectives

After viewing module "Server Load Balancing" and then completing this lab, you should be able to:

- Configure basic server load balancing (SLB).
- Export (save) SLB configurations to a file.
- Validate your configuration with test-traffic to web servers.
- Create advanced health checks for multiple services.
- Load balance to multiple application ports.

# Overview

In this lab we configure the Alteon to support server load balancing.
We configure server load balancing (SLB) by setting up Layer 4 real servers and binding them into a group.

We also bind the load balancing metric and health check to the server group. Then we configure the virtual server and assign a service -- and bind that the server group. We are all set to load balance.

Advanced health checks add flexibility in determining the condition of servers.

In this lab we create different health checks for multiple services running on the servers. We set up dependencies. After we add multiple ports, we load balance between those ports.

## Lab Preparations: Restore Standard Setup

Before you begin this lab:

a. You should have successfully completed routing and switching STANDARD SETUP configuration.
b. You should have successfully completed High Availability configuration.
c. Access Alteon management port and login. Import HIGH AVAILABILITY configuration -- else be sure it is already the Alteon configuration.
d. Verify your current HIGH AVAILABILITY configuration is working before going forward.
e. All changes done at one Alteon, preferred Device A, and synced to the second HA device.

## Lab Activities

Here is a summary of what you'll be doing in lab:

1. **Configure basic server load balancing**
   - Define real servers
   - Create server group
   - Bind health check to the group
   - Create virtual server and bind to the group
   - Define virtual service

2. **Configure advanced health check**
   - Create advanced health checks
   - Bind HC to server
   - Group health check
   - Combine health checks
   - Add ports to a server

3. **Validate your configuration**

## Configuration Details

**Recommendation**: When you perform implementation in real life, be sure to collect information in preparation of your work.

| Real Servers | |
|---|---|
| **Real Server ID** | **Server IP Address** |
| WebServer1 | 10.200.1.100 |
| WebServer2 | 10.200.1.200 |

| Group | |
|---|---|
| **Group ID** | Group1 |
| **Servers** | WebServer1, WebServer2 |
| **Healthcheck** | ICMP |
| **Metric** | Round Robin |

| Virtual Server Settings | |
|---|---|
| **Virtual Server** | VIRT ID = Virt1 |
| **VIP Address** | 192.168.175.50 |
| **Floating IP** | 10.200.1.254 |

| Virtual Services | |
|---|---|
| **Service** | http (port 80) |
| **Proxy IP** | 10.200.1.15/32 |
| **Group** | Group ID = Group1 |
| **Binding** | Force Proxy |

# Configure Basic SLB

## Define real servers

1. Define real WebServer1 and WebServer2.

| Configure Real Servers | | |
|---|---|---|
| **Real Server ID** | **Server IP Address** | **Description** |
| WebServer1 | 10.200.1.100 | Web Application Server 1 |
| WebServer2 | 10.200.1.200 | Web Application Server 2 |

Real server ID = up to 32 alpha-numeric characters, case-sensitive, NO SPACES.
Description is not important for load balancing. But, in case of debugging it is mostly very helpful!
NOTE: After each "**Apply**" the Alteon A will ask if you want to synchronize the configuration only n case you have not enabled Automatic Sync at Modules to Sync.

GUI:

a. **Configuration → Application Delivery → Server Resources → Real Servers**
b. Click **+** [to add]
c. Make sure Real Server is enabled
d. Add values for **Real Server ID, Description** and **Server IP Address**
e. **Submit** changes
f. Repeat setup for second server. Alternatively **mark first server**, click **Duplicate Real Server** button, and adjust parameter for **Real Server ID, Description** and **Server IP Address.**

## Create Server Group

2. Create server group. Use server group ID=Group1.
Add WebServer1 and WebServer2 to the group.
Set metric for the group to roundrobin.

a. **Configuration → Application Delivery → Server Resources → Server Groups**
b. Click **+** [add a group]
c. **Group ID** Group1
d. **Add WebServer1** and **WebServer2** to selected for this group
e. **Group Settings** tab:
   a. **SLB Metric** = Round Robin
f. **Submit**
g. **Apply**

All real servers for a specific service belong together in one group.

## Create a Virtual Server

3. Create virtual server ID = Virt1

| Virtual Server Settings | |
|---|---|
| **Virtual Server** | VIRT ID = Virt1 |
| **Description** | Web Application |
| **VIP Address** | 192.168.175.50 |

a. **Configuration → Application Delivery → Virtual Services**
b. Click **+** [to add]
c. Make sure you click "**Enable Virtual Server**"
d. Set **Virtual Server ID** Virt1
e. Add **Description** Web Application
f. Set **IP Address** 192.168.175.50
g. Click **Submit**

## Define a Virtual Service

4. Define a virtual service as http port 80 and bind to the group.
Force connection through Proxy IP in the http service.
Bind to server group ID = Group1.

| Virtual Services | |
|---|---|
| **Service** | http (port 80) |
| **Proxy IP** | 10.200.1.15 / 255.255.255.255 |
| **Group** | Group ID = Group1 |
| **Delayed Binding** | Disable |

- **Configuration → Application Delivery → Virtual Services**
  a. Select and edit virtual server Virt1
  b. At Virtual Services Click **+** [to add]
  c. **Service Port:** 80
  d. **Group ID:** Group1
  e. **Proxy IP** tab:
     a. **Client NAT Mode:** Address/Subnet
     b. **Client NAT IPv4 Address:** 10.200.1.15
     c. **Mask:** 255.255.255.255

This is the entry, listening or termination IP address for a specific Alteon service. For Radware Lab, forcing through proxy IP ensures connection through the PIP.

f. **Submit** and **Close**

g. **Apply**
h. **Save**
i. **Sync** in case you did not choose Automatic Sync

**IMPORTANT**: Radware Labs use AlteonOS v34.+. Starting with AlteonOS v30.0+, **server load balancing, port client processing, server processing, and proxy processing** are enabled by default. If your business uses an older version, you must enable these features separately – on each respective port.

## Synchronize the Configuration

If you didn't enable automatic synchronization in the HA lab, synchronize Alteon A to Alteon B by either:
Click "**Sync**" (if you haven't done so in the step above)

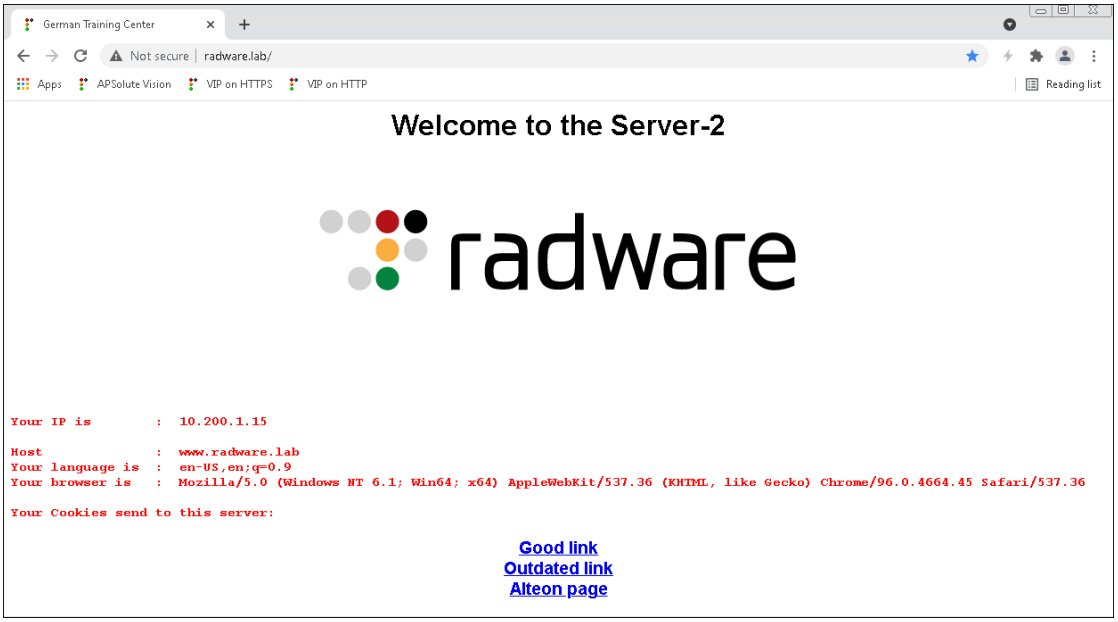
## Validate Configuration

### Verify configuration by generating test-traffic to your web servers.

Connect to virtual server IP (VIP) from your RDP PC. Browse to **http://radware.lab** or **http://192.168.175.50**

Using Chrome or Firefox allow viewing of web content within a different browser than doing the configuration task in Edge.

You should see a response that you reached WebServer1 or WebServer2. You may need to open a new browser window to see load balancing working.



### View status and statistics on your virtual server connection.

**GUI**: On Master Alteon!

- **Monitoring → Overview → Service Status View** (first time view take a couple of seconds)
- **Monitoring → Application Delivery → Server Resources → Real Servers**
- **Monitoring → Application Delivery → Server Resources → Server Groups**
- **Monitoring → Application Delivery → Virtual Servers** and select **virtual service**

**CLI**:

```
/info/slb/real WebServer1
/i/slb/real WebServer2
/i/slb/group Group1
/i/slb/virt Virt1
/stat/slb/virt Virt1
/st/slb/group Group1
/st/slb/real WebServer1
```

Get more detailed information on CLI:

```
/oper/displog ena
```

## Configure Advanced Health Check

### Create advanced health checks.

1. Create an advanced HTTP health check.
   a. Health ID = HTTP_Check
   b. Name = HTTP
   c. Path = index.htm
   d. Hostname = www.radware.lab

   **Configuration → Application Delivery → Server Resources → Health Check**
   a. Add a new Health Check (click **+** icon)
   b. Select Type **HTTP(S)** and **OK**
   c. **HTTP ID** HTTP_Check
   d. **Description** My HTTP health check
   e. **Destination Port** 80
   f. **HTTP(S) tab**:
   g. **Hostname** = www.radware.lab
   h. **Path** = index.htm
   i. **Submit**

2. Create a TCP port 81 health check.
   a. Select Type **TCP** and **OK**
   b. **TCP ID** TCP_Check
   c. **Description** My TCP_Check
   d. **Destination Port** 81
   e. **Submit**

3. Create an advanced HTTP health check for an image.
   Use path “images/img2.jpg” which ONLY resides on Server web2
   a. **Health ID** HTTP_IMG_Check
   b. **Description:** My_Image_check
   c. **Destination port** 80
   d. **Hostname** www.radware.lab
   e. **Path** /images/img2.jpg
   f. **Submit**
   g. **Apply**

## Bind Health Check to Group

Bind HTTP_Check health check to both real servers. (WebServer1 and WebServer2)

**Configuration → Application Delivery → Server Resources → Server Group**

a. **Edit** Group1
b. **Group Settings** tab
c. **Health Check** select HTTP_Check
d. **Submit**
e. **Apply**

**CLI:**

```
/i/slb/group Group1
Real Server Group Group1:
  metric roundrobin
  health HTTP_Check (HTTP), content
  maxconex: disabled
  Operation: enabled
  Virtual Server: Virt1, IP4 192.168.175.50
     Virtual Services:
       http: vport http, srvdown drop
         Real Servers:
         WebServer1: 10.200.1.100, Web Application Server 1, 00:0c:29:a9:4b:5c, vlan 14,
port 2, health port 80(runtime HTTP), 8 ms, UP

         WebServer2: 10.200.1.200, Web Application Server 2, 00:0c:29:da:75:c3, vlan 14,
port 2, health port 80(runtime HTTP), 8 ms, UP
```

Both servers should show up. Health check should say HTTP_Check.

Repeat above for:

- TCP_Check
  - Health check should show TCP_Check and both servers should be up.
- HTTP_IMG_Check
  - Health check should show HTTP_IMG_Check and WebServer1 should show FAILED.

## Combine Heath Checks

Create a health check to combine HTTP and TCP.
Use LOGEXP ID = HTTPTCP Type = LOGEXP Description = Combo_httptcp

**Configuration → Application Delivery → Server Resources → Health Check**

a. **Add a new Health Check** LOGEXP
b. **OK**
c. **LOGEXP ID**: HTTPTCP
d. **Description**: Combo_httptcp
e. From the **Health Check ID** drop down list **select** HTTP_Check
f. Enter the **&** in the Logical Expression, no spaces before/after the logical expression
g. From the **Health Check ID** drop down list **select** TCP_Check
h. **Submit**
i. **Apply**

Attach this combined health check to group Group1 and apply.

**Configuration → Application Delivery → Server Resources → Server Groups**

a. **Edit** Group1
b. **Group Setting** tab
c. **Health Check**: HTTPTCP
d. **Submit**
e. **Apply & Save**

Check the status of group Group1

**Configuration → Overview → Service Status View**

You should see that for each server the health module has created a runtime instance of each check of the logical expression.

**CLI:**

```
>> Alteon-A - Server Load Balancing Information# /i/slb/group Group1
Real Server Group Group1:
  metric roundrobin
  health HTTPTCP (LOGEXP), content
  maxconex: disabled
  Operation: enabled
  Virtual Server: Virt1, IP4 192.168.175.50
     Virtual Services:
       http: vport http, srvdown drop
         Real Servers:
         WebServer1: 10.200.1.100, Web Application Server 1, 00:0c:29:a9:4b:5c, vlan 14, port 2, health port
80(runtime LOGEXP), 3 ms, UP
            Expression string: HTTP_Check&TCP_Check

            HC instance HTTP_Check, type HTTP, name My HTTP health check, STATUS UP, 5 ms, Invert-result disabled

            HC instance TCP_Check, type TCP, name My TCP Check, STATUS UP, 2 ms, Invert-result disabled

         WebServer2: 10.200.1.200, Web Application Server 2, 00:0c:29:da:75:c3, vlan 14, port 2, health port
80(runtime LOGEXP), 2 ms, UP
            Expression string: HTTP_Check&TCP_Check

            HC instance HTTP_Check, type HTTP, name My HTTP health check, STATUS UP, 3 ms, Invert-result disabled

            HC instance TCP_Check, type TCP, name My TCP Check, STATUS UP, 2 ms, Invert-result disabled


Nov 17 10:36:14 NOTICE  slb: real server WebServer2, IP 10.200.1.200 operational
```

Change the group health check to TCP.

**Configuration → Application Delivery → Server Resources → Server Groups**

a. **Edit** Group1
b. **Group Settings** tab
c. **Health Check**: tcp
d. **Submit**
e. **Apply**

## Add multiple ports to the servers.

WebServer1 is running two HTTP instances listening to port 80 & 81, WebServer2 listen to port 80 – 83. Due to double number of instances set WebServer2 a weight value of 2 getting for each instance a similar amount of traffic.

**Configuration → Application Delivery → Server Resources → Real Servers**

- Edit WebServer1
- Add a **Service Port** using the **+** above the table called Service Port
- 80 and Submit
- 81 and Submit
- Submit and Close
- Edit WebServer2
- Add port 80 and Submit
- Add port 81 and Submit
- Add port 82 and Submit
- Add port 83 and Submit
- In the **Properties** tab, change the **Weight** to 2.
- Submit

Change Group Port Metric

Set group metric to Least Connections. Set rport metric to roundrobin.

**Configuration → Application Delivery → Server Resources → Server Groups**

a. **Edit** Group1
b. **Group Settings** tab
c. **SLB Metric**: Least Connections
d. **Advanced** tab
e. **Real Port Metric**: keep Round Robin
f. **Submit**

Change Virtual Server

Set the virtual server to use all the ports on the server by setting the Rport to 0.

**Configuration → Application Delivery → Virtual Services**

a. Select **Virtual Server** Virt1
b. Select **Virtual Services** of Selected Virtual Servers Virt1
c. **Edit** Virt1
d. **Properties** tab
e. Change **Real server Port** from 80 to 0
f. **Submit**
g. **Apply**

Use the CLI below to see the status of the virtual server. A confirmation line is printed for each port to be up.

```
Nov 17 10:57:17 INFO    mgmt: Operational change made by admin from WBM:10.10.240.15

Nov 17 10:57:17 INFO    HA: Configuration has been synchronized to # appliance 10.200.1.12

Nov 17 10:57:24 NOTICE  slb: real service WebServer1, IP 10.200.1.100:81 operational, affected virt 192.168.175.50

Nov 17 10:57:25 NOTICE  slb: real service WebServer2, IP 10.200.1.200:81 operational, affected virt 192.168.175.50

Nov 17 10:57:26 NOTICE  slb: real service WebServer2, IP 10.200.1.200:82 operational, affected virt 192.168.175.50

Nov 17 10:57:28 NOTICE  slb: real service WebServer2, IP 10.200.1.200:83 operational, affected virt 192.168.175.50
```

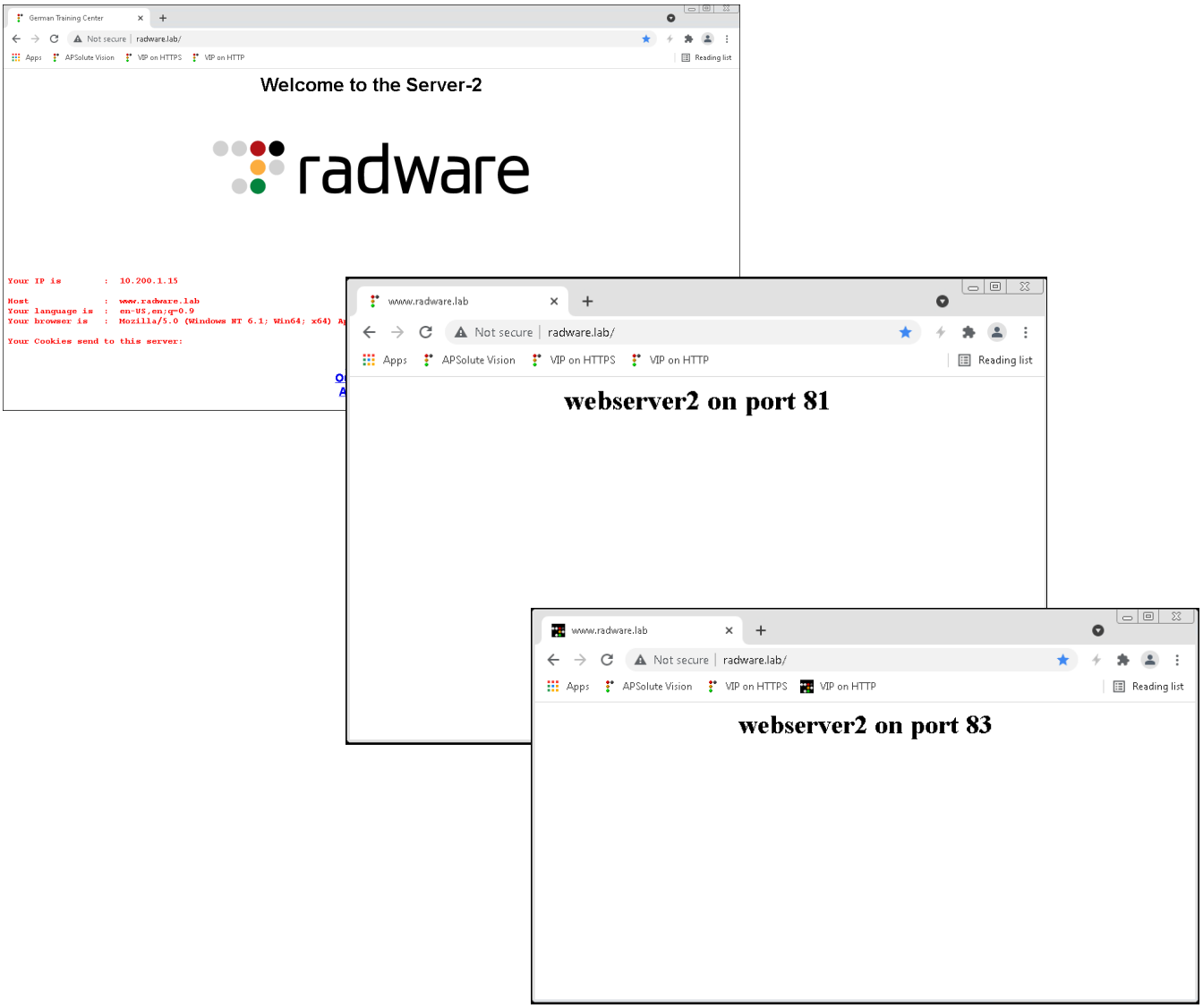
Use the command **/info/slb/virt Virt1** to see the status of the virtual server.

```
/i/slb/virt Virt1
Virt1: IP4 192.168.175.50,  00:03:b2:78:01:00, vname Web Application
    Virtual Services:
    http: rport 0, group Group1, health tcp (TCP), ipheader x-forwarded-for
        Real Servers:
        WebServer1: 10.200.1.100:80, Web Application Server 1, group ena, health  (runtime TCP), 1 ms, UP

        WebServer1: 10.200.1.100:81, Web Application Server 1, group ena, health  (runtime TCP), 1 ms, UP

        WebServer2: 10.200.1.200:80, Web Application Server 2, group ena, health  (runtime TCP), 1 ms, UP

        WebServer2: 10.200.1.200:81, Web Application Server 2, group ena, health  (runtime TCP), 2 ms, UP

        WebServer2: 10.200.1.200:82, Web Application Server 2, group ena, health  (runtime TCP), 0 ms, UP

        WebServer2: 10.200.1.200:83, Web Application Server 2, group ena, health  (runtime TCP), 1 ms, UP
```

Test Configuration

Open Chrome Browser and surf to http://www.radware.lab. You can use the bookmark VIP on HTTP.
Close the browser and open it again. Click on bookmark VIP on HTTP. SLB select the different pages available for these different web ports.

Before continuing with next hands-on lab **revert back Real server Port** from 0 **to 80**.



For questions, contact training@Radware.com