



**Radware's AppDirector  
And  
Microsoft Exchange 2010  
Integration Guide**

**Products:**

**Radware AppDirector**

**Software:** AppDirector version 2.14.00

**Version – 2.07**

## Contents

Joint Solution Overview .....	3
Microsoft Exchange 2010 Overview .....	3
Radware AppDirector Overview.....	3
Exchange 2010 Definitions .....	5
Understanding Load Balancing in Exchange 2010 .....	6
AppDirector and Microsoft Office Communication Server Architecture.....	7
Diagram 1.0 - AppDirector and Microsoft Exchange 2010 Reference Architecture (One Armed) ....	7
Primary Front-End AppDirector Configuration .....	8
Network Configuration .....	9
Farm Configuration.....	9
Servers Configuration .....	10
HTTP Policy Configuration .....	11
SSL Policy Configuration .....	12
Cache Policy Configuration .....	12
Compression Policy Configuration .....	12
Layer 7 Configuration .....	13
Layer 4 Configuration .....	14
AppDirector Health Monitoring.....	15
NAT Configuration .....	17
General Redundant Configuration Notes.....	17
Primary AppDirector VRRP Configuration.....	17
Mirroring Configuration .....	18
Auto-Generate the Backup AppDirector Configuration .....	19
Setting up basic IP connectivity on the Backup AppDirector .....	19
Auto Generating the Backup Configuration from the Primary AppDirector .....	19
Upload the Backup Configuration file to the device .....	19
Appendices .....	21
Appendix 1 – SSL Configuration for CAS servers .....	21
Appendix 2 – Exchange CAS Array Configuration .....	22
Appendix 3 – HTTP redirect to HTTPS.....	23
Appendix 4 – RPC Client Access .....	25
Appendix 5 – POP3 and IMAP4.....	29
Appendix 6 – Backend Encryption .....	36

## **Joint Solution Overview**

The Radware and Microsoft Exchange 2010 joint solution ensures Exchange 2010 customers solution resilience, efficiency and scale. Radware's AppDirector guarantees Exchange 2010 services maximum availability, scalability, performance and security. Managing the advanced messaging functionality in Exchange 2010, AppDirector provides advanced health monitoring to avoid system down time and advanced traffic management to deliver a best of breed subsystem. With a pay as you grow platform licensing model, AppDirector ensures long term investment protection facilitating incremental growth demanded by today's business.

## **Microsoft Exchange 2010 Overview**

Now, more than ever, your organization requires cost-effective and flexible communication tools. With Microsoft Exchange Server 2010 you can achieve new levels of reliability and performance with features that simplify your administration, help protect your communications, and delight your users by meeting their demands for greater mobility.

Microsoft Exchange Server, the cornerstone of Microsoft's Unified Communications solution, is a flexible and reliable messaging platform that can help you lower your messaging costs by 50-80%, increase productivity with anywhere access to business communications, and safeguard your business with protection and compliance capabilities that help you manage risk.

For more information visit Microsoft Exchange 2010 web page:  
<http://www.microsoft.com/exchange/en-us/overview.aspx>

## **Radware AppDirector Overview**

AppDirector uses advanced Layer 4-7 policies and granular application intelligence for end-to-end business-smart networking. AppDirector aligns server infrastructure operations with application front end requirements to eliminate

- Traffic surges
- Server bottlenecks
- Connectivity disconnects
- Downtime

This assures application access, full application continuity and redundancy. AppDirector enables fine tuning of network behavior based on granular application-specific classification of packets to optimize traffic flows for a wide range of enterprise applications such as Microsoft, Oracle, BEA, IBM, SAP and

other web-based applications including support for VoIP, streaming media and secure LDAP applications.

For more information visit Radware AppDirector web page:

<http://www.radware.com/Products/ApplicationDelivery/AppDirector/default.aspx>

## **Exchange 2010 Definitions**

### ***Microsoft Outlook***

Microsoft Outlook is a personal information manager from Microsoft. It can be used as a stand-alone application, or can work with Microsoft Exchange Server and Microsoft Office SharePoint Server for multiple users in an organization, such as shared mailboxes and calendars, Exchange public folders, SharePoint lists and meeting schedules. There are third-party add-on applications that integrate Outlook with devices such as BlackBerry mobile phones and with other software like Office & Skype internet communication. Developers can also create their own custom software that works with Outlook and Office components using Microsoft Visual Studio. In addition, Windows Mobile devices can synchronize almost all Outlook data to Outlook Mobile.

### ***Outlook Anywhere***

Outlook Anywhere utilizes the RPC Proxy component in Windows to proxy RPC calls to the RPC Client Access Service and Exchange Address Book Service.

### ***Outlook Web App (OWA)***

Outlook Web App is a webmail service of Microsoft Exchange Server 5.0 and later. The web interface of Outlook Web App resembles the interface in Microsoft Outlook. Outlook Web App comes as a part of Microsoft Exchange Server.

Outlook Web App lets you access your e-mail from any Web browser. Outlook Web App (known as Outlook Web Access in earlier versions of Microsoft Exchange) has been redesigned in Exchange 2010. Features such as Chat, Text Messaging, mobile phone integration, and Conversation View provide an enhanced user experience from any computer that has a Web browser. In Exchange Server 2010, these features can be accessed from an expanded set of Web browsers including versions of Internet Explorer later than 6.0, Firefox, Safari, and Google's Chrome.

### ***Exchange ActiveSync (EAS)***

Exchange ActiveSync is used by mobile devices to synchronize mailbox content with an Exchange server 2010. You can synchronize e-mail, contacts, calendar information, and tasks.

If you use a phone that has Windows Mobile 5.0 with the Messaging Security and Feature Pack (MSFP) installed or a later version, your mobile phone will support Direct Push. Direct Push technology is built into Exchange ActiveSync and keeps a mobile phone continuously synchronized with an Exchange mailbox.

## ***Exchange Web Services (EWS)***

Exchange Web Services is a web services application programming interface (API) that can be used by 3rd party applications to access mailbox data. It is also used by various Microsoft produced applications and devices for integration with Exchange, for example Outlook 2007 and later, Entourage 2008 for Macintosh (Web Services Edition), Office Communicator, and the Office Communicator Phone.

## ***Exchange Control Panel (ECP)***

Exchange Control Panel - Administrators can use the Exchange Control Panel for Outlook Web App to manage some on-premises tasks. The following is a list of the administrative features available:

- Text messaging integration
- Voice messaging integration
- Multiple mailbox search
- Additional proxy addresses for mailboxes
- Moderation and approval for distribution list submission

In addition, users have self-service capabilities in that they can perform administrative tasks via the Exchange Control Panel. The ECP enables users to perform common tasks without having to call the help desk.

## ***Remote Powershell***

Remote Powershell is the administrative interface that enables you to manage your Microsoft Exchange Server 2010 organization from the command line.

## **Understanding Load Balancing in Exchange 2010**

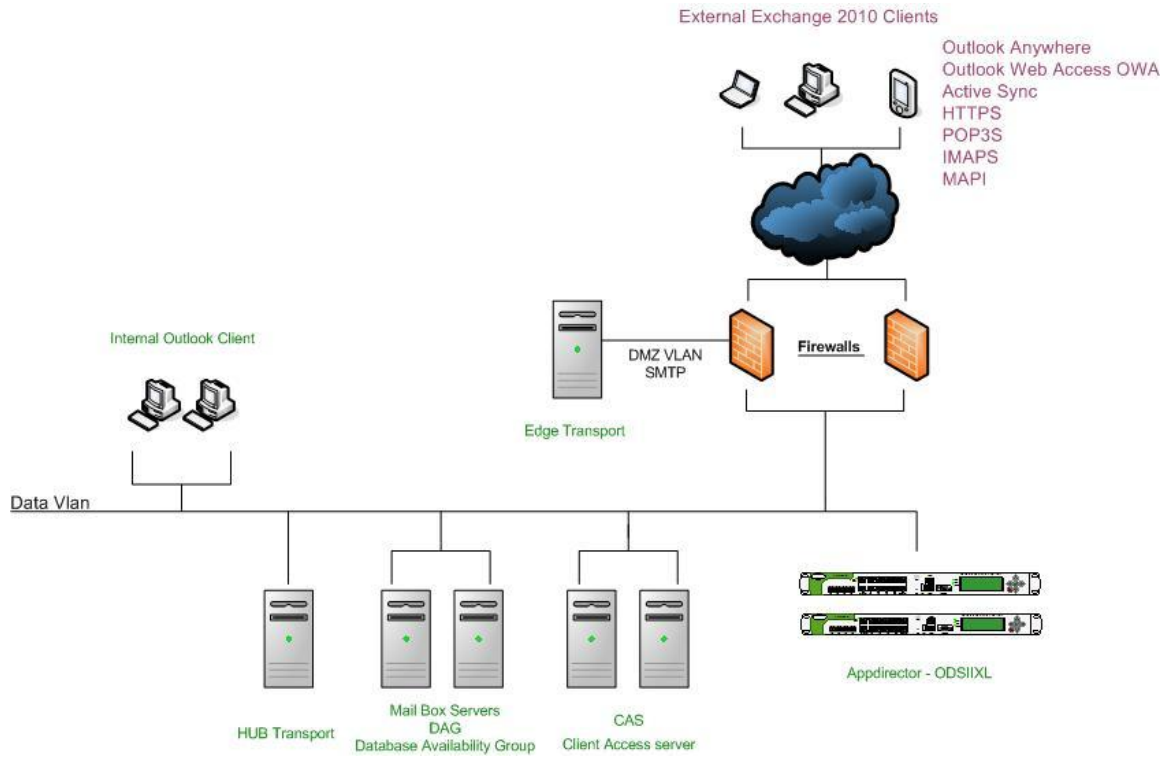
Load balancing is a way to manage which of your servers receive traffic. Load balancing provides failover redundancy to ensure your users continue to receive Exchange service in case of computer failure. It also enables your deployment to handle more traffic than one server can process while offering a single host name for your clients.

**Note:** For more detailed information please refer to:

<http://technet.microsoft.com/en-us/library/ff625247.aspx>

<http://technet.microsoft.com/en-us/library/ff625248.aspx>

# AppDirector and Microsoft Office Communication Server Architecture



**Diagram 1.0 - AppDirector and Microsoft Exchange 2010 Reference Architecture (One Armed)**

## Primary Front-End AppDirector Configuration

Using a serial cable and a terminal emulation program, connect to the AppDirector.

The default console port settings are:

- Bits per Second: 19200
- Data Bits: 8
- Parity: None
- Stop Bits: 1
- Flow Control: None

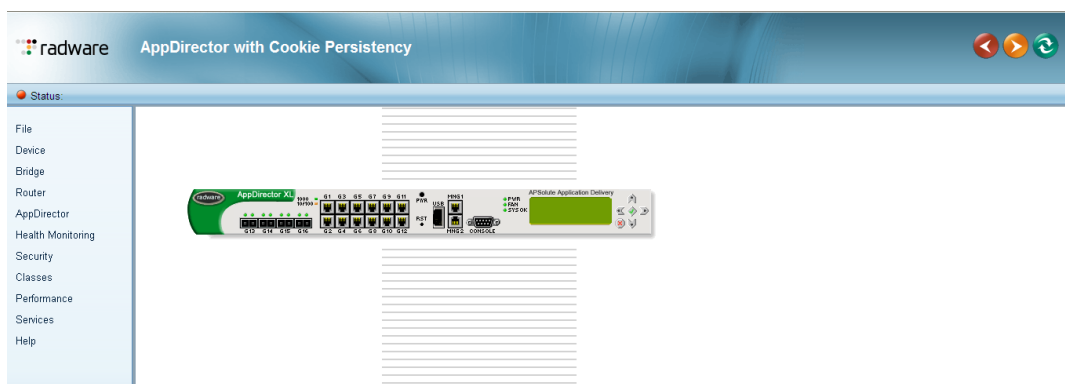
1. Using the following Command line, assign the management IP address 192.168.1.50 / 24 to interface MNG-1 (Dedicated Management Interface) of the AppDirector:

```
net ip-interface create 192.168.1.50 255.255.255.0 MNG-1 -pa 192.168.1.51
```

2. Using a browser, connect to the management IP Address of the AppDirector (192.168.1.50) via HTTP or HTTPS. The default username and password are “radware” and “radware”.

Failure to establish a connection may be due to the following:

- Incorrect IP Address in the browser
- Incorrect IP Address or default route configuration in the AppDirector
- Failure to enable Web Based Management or Secure Web Based Management in the AppDirector
- If the AppDirector can be successfully pinged, attempt to connect to it via Telnet or SSH. If the pinging or the Telnet/SSH connection are unsuccessful, reconnect to the AppDirector via its console port.



## ***Network Configuration***

1. Configure the AppDirector interface G1 for IP 10.1.3.201/16 in **Router -> IP Router -> Interface Parameters** with these parameters:
  - IP Address = 10.1.3.201
  - Network Mask = 255.255.240.0
  - Interface = G1
  - Peer Address = 10.1.3.202
2. Configure the AppDirector default route for IP 10.1.2.254 in **Router -> Routing** with these parameters:
  - Next Hop = 10.1.2.254
  - Interface = G1

## ***Farm Configuration***

We will create three farms for OWA, Outlook Anywhere and SMTP transport mail. Two farms will handle HTTP traffic (one for browsers that support cookies and one for browsers that don't) and one farm for SMTP traffic.

Outlook Anywhere for Exchange 2010 allows you to use Outlook 2003, Outlook 2007 and Outlook 2010 clients to connect to your Exchange server over the Internet, using HTTPS to encapsulate RPC traffic.

1. Create two farms named "**Webmail**" and "WebmailNoCookie" in **AppDirector -> Farms -> Farm Table** with these parameters:
  - Farm Name = Webmail
  - Aging Time = 3600
  - Session mode = EntryPerSession
  - Dispatch Method = Cyclic
  - Connectivity checks = No Checks
  - Leave all other fields as default
  - Farm Name = WebmailNoCookie
  - Aging Time = 3600
  - Session mode = EntryPerSession
  - Dispatch Method = Cyclic
  - Connectivity checks = No Checks
  - Leave all other fields as default
2. Update advanced settings for each of "**Webmail**" and "WebmailNoCookie" Farm in **AppDirector -> Farms -> Extended Parameters**.
  - Client NAT Address Range = 10.1.9.1

3. Create a farm named **“SMTP”** to handle traffic for the Hub Transport in **AppDirector -> Farms -> Farm Table** with these parameters:
  - Farm Name = SMTP
  - Aging Time = 60
  - Session mode = EntryPerSession
  - Dispatch Method = Cyclic
  - Connectivity checks = No Checks
  - Leave all other fields as default
  
4. Set the Client NAT address range for **“SMTP”** Farm in **AppDirector -> Farms -> Extended Parameters**.
  - Client NAT Address Range = 10.1.9.1

### ***Servers Configuration***

Note: The default and recommended configuration is to use SSL offloading and configure the servers to run on port 80. See Appendix 1 for instructions on how to configure IIS and the CAS servers with SSL offloading

1. Create a server named **“Webmail-Server1”** and attach it to the farm **“Webmail”** in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = Webmail
  - Server Address = 10.106.2.52
  - Server Name = Webmail-Server1
  - Server Port = 80
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default
  
2. Create a server named **“Webmail-Server2”** and attach it to the farm **“Webmail”** in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = Webmail
  - Server Address = 10.106.2.53
  - Server Name = Webmail-Server2
  - Server Port = 80
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default

3. Create a server named “**SMTP-Server1**” and attach it to the farm “SMTP” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = SMTP
  - Server Address = 10.106.2.54
  - Server Name = SMTP-Server1
  - Server Port = 25
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default
  
4. Create a server named “**SMTP-Server2**” and attach it to the farm “SMTP” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = SMTP
  - Server Address = 10.106.2.55
  - Server Name = SMTP-Server2
  - Server Port = 25
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default
  
5. Create a server named “WebmailNoCookie-Server1” and attach it to the farm “WebmailNoCookie” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = WebmailNoCookie
  - Server Address = 10.106.2.52
  - Server Name = WebmailNoCookie-Server1
  - Server Port = 80
  - Leave all other fields as default
  
6. Create a server named “WebmailNoCookie-Server2” and attach it to the farm “WebmailNoCookie” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = WebmailNoCookie
  - Server Address = 10.106.2.53
  - Server Name = WebmailNoCookie-Server2
  - Server Port = 80
  - Leave all other fields as default

### ***HTTP Policy Configuration***

Outlook Anywhere uses Basic Authentication by default. If you use Basic Authentication, you are able to take advantage of HTTP Multiplexing (serving multiple clients over much smaller number of server connections).

To configure HTTP Multiplexing create a new HTTP Policy under **AppDirector -> L4 Traffic Redirection -> HTTP Policy** with these parameters:

- Policy name = Exchange\_HTTP\_policy
- Multiplex Back-End connections = Enabled
- Leave all other fields as default

### ***SSL Policy Configuration***

**Note:** First configure the Outlook Web App servers to support SSL offloading, please follow the Microsoft documentation.

**Note:** Configures in the SSL policy a reference is made to the “radware” pre-configured certificate, but you can import a certificate or create a new certificate in AppDirector. For more information on exporting, importing, or creating a certificate, see the *AppDirector User Guide*.

Create an SSL policy in **AppDirector -> L4 Traffic Redirection -> SSL Policy** with these parameters:

- Policy name = Exchange\_SSL\_policy
- Certificate = radware
- Listening Server Port = 80 (servers listen on this port)
- Backend SSL State = Disabled
- Leave all other fields as default

### ***Cache Policy Configuration***

Create a Cache policy in **AppDirector -> L4 Traffic Redirection -> Cache Policies** with these parameters:

- Policy name = Exchange\_Cache
- Leave all other fields as default

### ***Compression Policy Configuration***

Create a Cache policy in **AppDirector -> L4 Traffic Redirection -> Compression Policies** with these parameters:

- Policy name = Exchange\_Compression
- Algorithm = Gzip
- Engine = Software (if you have Hardware compression card please choose Hardware)
- Leave all other fields as default

### **Layer 7 Configuration**

Under Layer 7 Method Table Create a Layer 7 Method for OWA and Outlook Anywhere Traffic.

1. Create a Layer 7 Method for Outlook Anywhere Traffic named “Default” in **AppDirector -> Layer 7 Farm Selection -> Methods** with these parameters:
  - Method Name = Default
  - Method Type = Regular Expression
  - Arguments
    - Regular Expression = . (dot)
  - Leave all other fields as default
  
2. Create a Layer 7 Method for OWA Traffic named “OutlookSessionCookie” in **AppDirector -> Layer 7 Farm Selection -> Methods** with these parameters:
  - Method Name = OutlookSessionCookie
  - Method Type = Cookie
  - Arguments
    - Key = OutlookSession
  - Leave all other fields as default

Under Layer 7 Policy Table Create a Layer 7 Policy for OutlookAnywhere and Outlook Anywhere Traffic.

3. Create a Layer 7 policy for OWA and Outlook Anywhere Traffic named “OutlookAnywhere” in **AppDirector -> Layer 7 Farm Selection -> Policies** with these parameters:
  - Policy Name = OutlookAnywhere
  - Policy Index = 10
  - First Method = OutlookSessionCookie
  - Farm Name = Webmail
  - Leave all other fields as default
  
4. Create a second Layer 7 policy entry for OWA and Outlook Anywhere Traffic named “OutlookAnywhere” in **AppDirector -> Layer 7 Farm Selection -> Policies** with these parameters:
  - Policy Name = OutlookAnywhere
  - Policy Index = 20

- First Method = Default
- Farm Name = WebmailNoCookie
- Leave all other fields as default

Under Layer 7 Server Persistency Text Match Create Layer 7 Persistency policies for Outlook Anywhere Traffic. For Microsoft Outlook client who do not support cookies, persist based on the “Authorization” HTTP header Information. In case of NTLM authentication, only basic Client IP persistency (L3) is used.

5. Create a Layer 7 Text Match Session ID Persistency for Outlook Anywhere Traffic named “Webmail” in **AppDirector -> Layer 7 Server Persistency -> Text Match** with these parameters:
  - Farm Name = Webmail
  - Application Port = 443
  - Lookup Mode = Cookie
  - Persistency Parameter = OutlookSession
  - Learning Direction = Client Request
  - Inactivity Timeout = 3600
  - Leave all other fields as default
  
6. Create a Layer 7 Text Match Session ID Persistency for Outlook Anywhere Traffic named “WebmailNoCookie” in **AppDirector -> Layer 7 Server Persistency -> Text Match** with these parameters:
  - Farm Name = WebmailNoCookie
  - Lookup Mode = Header
  - Header Name = Authorization
  - Learning Direction = Client Request
  - Inactivity Timeout = 3600
  - Leave all other fields as default

#### ***Layer 4 Configuration***

1. Create a Layer 4 policy for SMTP Traffic named “smtp\_policy” in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:
  - Virtual IP = 10.1.3.152
  - L4 Protocol = TCP
  - L4 Port = 25
  - L4 Policy Name = smtp\_policy
  - Application = TCP
  - Farm Name = SMTP
  - Leave all other fields as default

2. Create a Layer 4 policy for HTTPS Traffic named "Webmail-https" in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:

- Virtual IP = 10.1.3.152
- L4 Protocol = TCP
- L4 Port = 443
- L4 Policy Name = Webmail-https
- Application = HTTPS
- Farm Name = None
- L7 Policy Name = OutlookAnywhere
- HTTP Policy = Exchange\_HTTP\_policy
- SSL Policy = Exchange\_SSL\_policy
- Compression Policy = Exchange\_Compression
- Caching Policy = Exchange\_Cache
- Leave all other fields as default

### ***AppDirector Health Monitoring***

1. Enable Health Monitoring in **Health Monitoring -> Global Parameters**.

Health Check

2. Create a check for Webmail\_svr1 server IP address 10.106.2.52 in **Health Monitoring -> Check Table**:

- Check name = Webmail\_svr1
- Method = HTTP
- Dest IP = 10.106.2.52
- Dest Port = 80
- Interval = 91
- Timeout = 30
- Leave all other fields as default

3. Create a check for Webmail\_svr2 server IP address 10.106.2.53 in **Health Monitoring -> Check Table**:

- Check name = Webmail\_svr2
- Method = HTTP
- Dest IP = 10.106.2.53
- Dest Port = 80
- Interval = 91
- Timeout = 30
- Leave all other fields as default

4. Create a check for SMTP on server 10.106.2.54 in **Health Monitoring -> Check Table:**

- Check name = smtp\_hub1
- Method = SMTP
- Dest IP = 10.106.2.54
- Dest Port = 25
- Interval = 91
- Timeout = 30
- Leave all other fields as default

5. Create a check for SMTP on server 10.106.2.55 in **Health Monitoring -> Check Table:**

- Check name = smtp\_hub2
- Method = SMTP
- Dest IP = 10.106.2.55
- Dest Port = 25
- Interval = 91
- Timeout = 30
- Leave all other fields as default

Health Check Binding

6. Bind the check Webmail\_svr1 to Farm Webmail - 10.106.2.52 - 80 in **Health Monitoring -> Binding Table.**

7. Bind the check Webmail\_svr2 to Farm Webmail - 10.106.2.53 - 80 in **Health Monitoring -> Binding Table.**

**Note:** Since we are using the same servers and HTTP for both OWA and Outlook Anywhere and run the same health check, we can bind the Webmail check to the WebmailNoCookie servers or you can create a separate check for these servers.

8. Bind the check Webmail\_svr1 to Farm WebmailNoCookie - 10.106.2.52 - 80 in **Health Monitoring -> Binding Table.**

9. Bind the check Webmail\_svr2 to Farm WebmailNoCookie - 10.106.2.53 - 80 in **Health Monitoring -> Binding Table.**

10. Bind the check smtp\_hub1 to Farm SMTP - 10.106.2.52 – 25 in **Health Monitoring -> Binding Table.**

11. Bind the check smtp\_hub2 to Farm SMTP - 10.106.2.53 – 25 in **Health Monitoring -> Binding Table.**

## ***NAT Configuration***

1. Enable Client NAT from Global Parameters in **AppDirector -> NAT -> Client NAT -> Global Parameters**
2. Create the Client NAT intercept range in **AppDirector -> NAT -> Client NAT -> Intercept Addresses** with these parameters:
  - From Client IP = 1.1.1.1
  - To Client IP = 254.254.254.254
3. Create the Client NAT address range in **AppDirector -> NAT -> Client NAT -> NAT Addresses** with these parameters:
  - From Client IP = 10.1.9.1
  - To Client IP = 10.1.9.1

**Note:** You may need to add additional Client NAT addresses in order to scale to your client connection requirements. You can handle up to 65,000 connections per IP.

## **General Redundant Configuration Notes**

For complete high-availability, Radware encourages implementing pairs of AppDirector units in an Active / Backup configuration. If your implementation of this architecture includes only a single AppDirector, then it is unnecessary to follow the steps in this section.

### ***Primary AppDirector VRRP Configuration***

1. Enable VRRP in **Redundancy -> Global Configuration** with these parameters:
  - IP Redundancy Admin Status = VRRP
  - Interface Grouping = Enable
  - ARP with interface grouping = Send
  - Backup Fake ARP = Enable
  - Backup Interface Grouping = Enable
  - Leave all other fields as default
2. Create Virtual Router interfaces in **Redundancy -> VRRP -> Virtual Routers** with these parameters:
  - IF Index = G-1

- VR ID = 1
  - Priority = 255 (Highest number is Active device)
  - Primary IP = 10.1.3.201
  - Leave all other options as default
3. Create Associated IP Addresses in **Redundancy -> VRRP -> Associated IP Addresses** with these parameters:
    - IF Index – G-1, VR ID – 1, Associated IP 10.1.3.152 (VIP)
    - IF Index – G-1, VR ID – 1, Associated IP 10.1.3.201 (G-1)
    - IF Index – G-1, VR ID – 1, Associated IP 10.1.9.1 (Client NAT)
  4. Go to **Redundancy -> VRRP -> Virtual Routers** and on the **Virtual Router Table** under VRID's Up/Down select "All Up" and click on the **Set** button to enable all Virtual Routers.
  5. Make certain that the State of this VR is displayed as Master in the Virtual Router table.

### ***Mirroring Configuration***

1. Enable Mirroring in **Redundancy -> Mirroring -> Active Device Parameters** with these parameters:
  - Client Table Mirroring = Enable
  - Session Id Table Mirroring = Enable
  - Leave all other fields as default
2. Add Mirror device in **Redundancy -> Mirroring -> Mirror Device Parameters** with the following parameter:
  - Mirror Device IP = 10.1.3.202

Note: This sets the Backup AD target address used for mirror traffic.

## **Auto-Generate the Backup AppDirector Configuration**

Once the Backup AppDirector is configured for basic IP connectivity and is available to the network, simply export the Backup Configuration file from the Primary AppDirector and upload it to the Backup AppDirector. The steps are defined below.

### ***Setting up basic IP connectivity on the Backup AppDirector***

Using a serial cable and a terminal emulation program, connect to the AppDirector.

The default console port settings are:

- Bits per Second: 19200
  - Data Bits: 8
  - Parity: None
  - Stop Bits: 1
  - Flow Control: None
3. Using the following Command line, assign the management IP address 192.168.1.51 / 24 to interface MNG-1 (Dedicated Management Interface) of the AppDirector:

```
net ip-interface create 192.168.1.51 255.255.255.0 MNG-1 -pa 192.168.1.50
```

4. Using a browser, connect to the management IP Address of the AppDirector (192.168.1.51) via HTTP or HTTPS. The default username and password are “radware” and “radware”.

### ***Auto Generating the Backup Configuration from the Primary AppDirector***

1. From the web interface menu of the **Primary AppDirector**, select **File -> Configuration -> Receive from Device** to display the **Download Configuration File** page enter with these parameters:

- Configuration Type - Regular

2. On the **Configuration File Download** page, choose the necessary parameters as shown below:

- Configuration Type – Backup (Active-Backup)

3. Click the **Set** button to launch save file window.
4. Click the **SAVE** button to save the file to a local directory.

### ***Upload the Backup Configuration file to the device***

From the web interface menu of the **Backup AppDirector**, select **File -> Configuration -> Send to Device**. On the **Configuration File Upload** page choose the necessary parameters as shown below:

- Upload Mode – Replace configuration file
- Configuration file – Clicking the Browse button and navigate to the updated configuration file. Click the **Set** button to upload the configuration.

This completes redundancy configuration on the Backup AppDirector.

## **Appendices**

### **Appendix 1 – SSL Configuration for CAS servers**

#### **Exchange SSL Offload Settings**

As a requirement of SSL offload to the AppDirector, a minor change needs to be made to the CAS servers. When a SSL connection is terminated on the ADC, all connections from the AppDirector to the CAS servers are made via standard http.

Exchange server settings are detailed under <http://social.technet.microsoft.com/wiki/contents/articles/how-to-configure-ssl-offloading-in-exchange-2010.aspx>

## Appendix 2 – Exchange CAS Array Configuration

Another requirement to support hardware load balancing for your CAS servers is the configuration of a CAS array. In its configuration you must create a CAS array through the Exchange Management Shell that groups your CAS servers together. Then make sure the name of your CAS array is a DNS registered name that points to the AppDirector load balancer virtual IP.

1. On your domain controller, create a forward lookup entry that maps your AppDirector load balancer virtual IP to the fully-qualified domain name that you assign to your CAS array in the next step.
2. Then, create the CAS array with the command "New-ClientAccessArray -Name <NameoftheArray> -FQDN <NameoftheArray.Fully-qualified domain name> -Site <ADSiteName>".
3. For each of the mailbox databases that will be front-ended by your CAS servers, you need to assign the CAS array as the RPC Client Access server setting for each mailbox database. To see a list of all your mailbox database names, use the "Get-MailboxDatabase" command.
4. If you need to find out what each of your Mailbox databases is using as its CAS server, use the command "GET-MailboxDatabase "dbase4" | fl RpcClientAccessServer".
5. Set the CAS server setting for each mailbox database to be the newly created CAS array with the command "Set-MailboxDatabase <database name/id> -RpcClientAccessServer <NameoftheArray.Fully-qualified domain name>".
6. When you initiate a connection from your Exchange user, be sure to specify the CAS array fully-qualified domain name (which is the same as the AppDirector load balancer VIP fully-qualified domain name) as the server to which you are connecting. The connection then goes to the AppDirector to be load-balanced amongst the CAS servers in your CAS array.

## Appendix 3 – HTTP redirect to HTTPS

The following instructions shows how to create a L7 Policy that redirects HTTP traffic to same host name same URI over HTTPS. This L7 Policy is a safety net; it catches the traffic that incorrectly comes in on HTTP and redirects it to HTTPS.

https://[Host Name]/owa\*  
https://[Host Name]/rpc\*

### Methods Table

A method is defined to identify the Host/URI that is used to identify the traffic that is to be converted from HTTP to HTTPS.

1. Create a Layer 7 Method to identify the Host/URI in **AppDirector -> Layer 7 Farm Selection -> Methods** with these parameters:
  - Method Name = HTTP\_Redirect
  - Method Type = URL
  - Arguments
    - Host Name = <host name> (example = radware.com)
    - Path = /owa\*

Or

  - Path = /rpc\*
  - Leave all other fields as default

### Layer 7 Policy Table

2. Create a Layer 7 policy to redirect HTTP to HTTPS in **AppDirector -> Layer 7 Farm Selection -> Policies** with these parameters:
  - Policy Name = HTTPRedirectToHTTPS
  - Policy Index = 10
  - First Method = HTTP\_Redirect
  - Arguments
    - HTTPS Redirect TO = <host name> (example = radware.com)
  - Leave all other fields as default

**Note:** HTTPS Redirect to (RDRS): AppDirector redirects the HTTP request to the specified name or IP and modifies the request to a HTTPS request.

### Create Layer 4 Policy

3. Create a Layer 4 policy to select the L7 Policy that redirects HTTP to HTTPS Traffic for OWA and Outlook Anywhere in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:

- Virtual IP = 10.1.3.152
- L4 Port = 80
- L4 Policy Name = Exchange\_Redirect
- Application = HTTP
- Farm Name = None
- L7 Policy Name = HTTPRedirectToHTTPS
- Leave all other fields as default

## Appendix 4 – RPC Client Access

With Exchange Server 2010, Outlook clients connect using native MAPI to the new RPC Client Access service, which runs on Client Access servers, rather than directly to Mailbox servers.

**Note:** When you upgrade your organization to Exchange 2010, your clients running Outlook 2007 or later versions will automatically be compatible with the change to RPC Client Access, since they support RPC encryption by default. Outlook 2003 doesn't use RPC encryption, however, and RPC Client Access requires it by default. If you haven't turned off RPC encryption, your users will need to configure Outlook 2003 for RPC encryption or you'll need to use a Group Policy to force Outlook 2003 to use RPC encryption.

**Note:** Because the RPC Client Access Service requires the traffic to be passed to the Client Access servers on a large number of ports, we recommend that you use a firewall to permit only internal networks to access the RPC Client Access virtual server IP address.

### Configuring Static Port Mapping For RPC-Based Services

The static port for the RPC Client Access Service is configured via the registry. The following registry key should be set on each Client Access Server to the value of the port that you wish to use for TCP connections for this service.

Key:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeRPC\ParametersSystem  
Value: TCP/IP Port  
Type: DWORD

Note that this will only affect connections for “internal” connections via TCP and will not affect Outlook Anywhere connections that take advantage of RPC/HTTP tunneling. Outlook Anywhere connections to the RPC Client Access Service will occur on port 6001 and this is not configurable.

The static ports for the two RPC endpoints maintained by the Exchange Address Book Service are set in the Microsoft.Exchange.AddressBook.Service.Exe.config file which can be found in the bin directory under the Exchange installation path on each Client Access Server. The “RpcTcpPort” value in the configuration file should be set to the value of the port that you wish to use for TCP connections for this service. This port will handle connections for both the Address Book Referral (RFR) interface and the Name Service Provider Interface (NSPI).

N

ote that the values for the “NspiHttpPort” and “RfrHttpPort” configuration options should not be changed as Outlook is configured to use these ports by default. Changing these values may result in unwanted delay when attempting to establish Outlook Anywhere connections.

**Note:** For Exchange 2010 SP1 please refer to:

<http://social.technet.microsoft.com/wiki/contents/articles/configuring-static-rpc-ports-on-an-exchange-2010-client-access-server.aspx>

## RPC Client Access Configuration requirements

Please refer to Microsoft documentation regarding the configuration of the RPC Client Access service and mailboxes for each site. To work with a load balancer, complete the following steps.

1. In the Microsoft Exchange Management Shell, create a new Client Access Array and associate it with the same FQDN that you will be using.

example:

```
New-ClientAccessArray -Name "Internal Client Array" -FQDN outlook.radware.com
```

```
New-ClientAccessArray [-Name <String>] -Fqdn <Fqdn> -Site  
<AdSiteIdParameter> [-Confirm [<SwitchParameter>]] [-DomainController  
<Fqdn>] [-WhatIf [<SwitchParameter>]]  
New-ClientAccessArray -Name "Internal Client Array" -FQDN outlook.radware.com
```

2. You must modify the attributes of any pre-existing mailbox databases to use the new array.

```
Set-MailboxDatabase "Mailbox Database"-RpcClientAccessServer CAS-Array.radware.com
```

**Note:** You can only configure one Client Access Array (one FQDN) per site.

**Note:** In the configuration example below, the static port for the RPC Client Access Service is configured via the registry to port 135. Please see Deployment notes above for instructions on how to configure the registry. In case the default port is changed, make sure to change the port number from 135 to the new static port in all of the following configuration settings.

To load balance the RPC Client Access Service, you will need to create a new farm, add servers to the farm, create a new L4 policy for port 135 and configure health monitoring.

1. Create a farm named "**Client\_RPC\_Internal**" in **AppDirector -> Farms -> Farm Table** with these parameters:
  - Farm Name = Client\_RPC\_Internal
  - Aging Time = 3600
  - Session mode = EntryPerSession

- Dispatch Method = Cyclic
- Connectivity checks = No Checks
- Leave all other fields as default

Note: Persistency for RPC Client Access Service uses Source IP and is defined by the Session mode set to “EntryPerSession” in the Farm.

2. Create a server named “C\_Int\_RPC\_Svr1” and attach it to the farm “Client\_RPC\_Internal” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:

- Farm Name = Client\_RPC\_Internal
- Server Address = 10.106.2.52
- Server Name = C\_Int\_RPC\_Svr1
- Server Port = 135
- Client NAT = Enable
- Client NAT Address Range = 10.1.9.1
- Leave all other fields as default

3. Create a server named “C\_Int\_RPC\_Svr2” and attach it to the farm “Client\_RPC\_Internal” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:

- Farm Name = Client\_RPC\_Internal
- Server Address = 10.106.2.53
- Server Name = C\_Int\_RPC\_Svr2
- Server Port = 135
- Client NAT = Enable
- Client NAT Address Range = 10.1.9.1
- Leave all other fields as default

4. Create a Layer 4 policy to select RPC Client Access Service over port 135 in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:

- Virtual IP = 10.1.3.152
- L4 Port = 135
- L4 Policy Name = Internal\_Client\_RPC
- Application = Any
- Farm Name = Client\_RPC\_Internal
- Leave all other fields as default

5. Create a check RPC\_Internal\_Client\_Svr1 server IP address 10.106.2.52 in **Health Monitoring -> Check Table**:

- Check name = RPC\_Internal\_Client\_Svr1

- Method = TCP Port
  - Dest IP = 10.106.2.52
  - Dest Port = 135
  - Interval = 91
  - Timeout = 30
  - Leave all other fields as default
6. Create a check for RPC\_Internal\_Client\_Svr2 server IP address 10.106.2.53 in **Health Monitoring -> Check Table**:
- Check name = RPC\_Internal\_Client\_Svr2
  - Method = TCP Port
  - Dest IP = 10.106.2.53
  - Dest Port = 135
  - Interval = 91
  - Timeout = 30
  - Leave all other fields as default
7. Bind the check RPC\_Internal\_Client\_Svr1 to Farm Farm Client\_RPC\_Internal - 10.106.2.52 - 135 in **Health Monitoring -> Binding Table**.
8. Bind the check RPC\_Internal\_Client\_Svr2 to Farm Farm Client\_RPC\_Internal - 10.106.2.53 - 135 in **Health Monitoring -> Binding Table**.

## Appendix 5 – POP3 and IMAP4

POP3 and IMAP4 enable a variety of clients to connect to the Exchange server. These include Outlook, Outlook Express, and third-party clients such as Eudora or Mozilla Thunderbird.

By default, POP3 and IMAP4 are disabled in Microsoft Exchange Server 2010. To support clients that still rely on these protocols, you must first start the POP3 and IMAP4 services on the Exchange 2010 Client Access server. You must also configure SMTP for your POP3 and IMAP4 clients to send e-mail.

For detailed steps about how to enable the POP3 and IMAP4 services, see Links below:

Enable POP3 in Exchange 2010

<http://technet.microsoft.com/en-us/library/bb124934.aspx>

Enable IMAP4 in Exchange 2010

<http://technet.microsoft.com/en-us/library/bb124489.aspx>

For more information about how to manage POP3 and IMAP4 in Exchange 2010, see Understanding POP3 and IMAP4 on Microsoft TechNet at

<http://technet.microsoft.com/en-us/library/bb124107%28EXCHG.140%29.aspx>

The following section shows how to configure the secure versions of POP3 and IMAP4, known as POP3S and IMAPS.

### IMAP Configuration

By default, the Exchange 2010 IMAP4 service requires encrypted connections. Since AppDirector will be terminating the SSL Connection you must first change the default setting on each Client Access server. You can either change the default setting from the Exchange Management Console or the Management Shell.

#### To change the default setting using the Exchange Management Console

1. Expand Server Configuration, then Client Access.
2. In the list of Client Access servers, select a server to which you will be sending IMAP4 traffic.
3. Select the IMAP4 protocol, right-click, and select Properties.
4. On the Authentication tab, change the setting to one of the plain text login methods (Basic or Integrated Windows) as appropriate for your environment and clients.
5. Click OK.
6. Restart the IMAP4 service on that Client Access server.

7. Repeat for each of the Client Access servers to which you will be sending IMAP4 connections.

### To change the default setting using the Exchange Management Shell

1. Type one of the following commands, substituting the name of a Client Access server for “**servername**”:

For Basic authentication:

```
Set-ImapSettings -Server "servername" -LoginType PlainTextLogin
```

For Windows Integrated authentication

```
Set-ImapSettings -Server "servername" -LoginType PlainTextAuthentication
```

2. Restart the IMAP4 service on that Client Access server.
3. Repeat for each of the Client Access servers to which you will be sending IMAP4 connections.

To load balance the IMAP4 Service, you will need to create a new farm, add servers to the farm, create a new L4 policy for TLS port **993** to backend server port **143** and configure health monitoring

1. Create a farm named “**CAS\_IMAP**” in **AppDirector -> Farms -> Farm Table** with these parameters:

- Farm Name = CAS\_IMAP
- Aging Time = 30
- Session mode = EntryPerSession
- Dispatch Method = Cyclic
- Connectivity checks = No Checks
- Leave all other fields as default

**Note:** No Persistency required for IMAP4.

2. Create a server named “IMAP\_Svr1” and attach it to the farm “CAS\_IMAP” in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:

- Farm Name = CAS\_IMAP
- Server Address = 10.106.2.52
- Server Name = IMAP\_Svr1
- Server Port = 143
- Client NAT = Enable
- Client NAT Address Range = 10.1.9.1
- Leave all other fields as default

3. Create a server named "IMAP\_Svr2" and attach it to the farm "CAS\_IMAP" in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:
  - Farm Name = CAS\_IMAP
  - Server Address = 10.106.2.53
  - Server Name = IMAP\_Svr2
  - Server Port = 143
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default
  
4. Create an SSL policy in **AppDirector -> L4 Traffic Redirection -> SSL Policy** with these parameters:
  - Policy name = Exchange\_IMAP\_SSL\_policy
  - Certificate = radware
  - Listening Server Port = 143 (servers listen on this port)
  - Backend SSL State = Disabled
  - Leave all other fields as default
  
5. Create a Layer 4 policy to select IMAP4 Service over TLS port 993 in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:
  - Virtual IP = 10.1.3.152
  - L4 Port = 993
  - L4 Policy Name = IMAP\_TLS
  - Application = Generic-SSL
  - SSL Policy = Exchange\_IMAP\_SSL\_policy
  - Application = Any
  - Farm Name = CAS\_IMAP
  - Leave all other fields as default
  
6. Create a check IMAP\_Svr1 server IP address 10.106.2.52 in **Health Monitoring -> Check Table**:
  - Check name = IMAP\_Svr1
  - Method = IMAP4
  - Dest IP = 10.106.2.52
  - Dest Port = 143
  - Interval = 91
  - Timeout = 30
  - Arguments
    - Username = <the configured user name>
    - Password = <the configured password>

- Leave all other fields as default
7. Create a check for IMAP\_Svr2 server IP address 10.106.2.53 in **Health Monitoring -> Check Table**:
    - Check name = IMAP\_Svr2
    - Method = IMAP4
    - Dest IP = 10.106.2.53
    - Dest Port = 143
    - Interval = 91
    - Timeout = 30
    - Arguments
      - Username = <the configured user name>
      - Password = <the configured password>
    - Leave all other fields as default
  8. Bind the check IMAP\_Svr1 to Farm CAS\_IMAP - 10.106.2.52 - 143 in **Health Monitoring -> Binding Table**.
  9. Bind the check IMAP\_Svr2 to Farm CAS\_IMAP - 10.106.2.53 - 143 in **Health Monitoring -> Binding Table**.

## **POP Configuration**

By default, the Exchange 2010 POP service requires encrypted connections. Since AppDirector will be terminating the SSL Connection you must first change the default setting on each Client Access server. You can either change the default setting from the Exchange Management Console or the Management Shell.

### **To change the default setting using the Exchange Management Console**

1. Expand Server Configuration, then Client Access.
2. In the list of Client Access servers, select a server to which you will be sending POP3 traffic.
3. Select the POP3 protocol, right-click, and select Properties.
4. On the Authentication tab, change the setting to one of the plain text login methods (Basic or Integrated Windows) as appropriate for your environment and clients.
5. Click OK.
6. Restart the POP3 service on that Client Access server.
7. Repeat for each of the Client Access servers to which you will be sending POP3 connections.

### **To change the default setting using the Exchange Management Shell**

1. Type one of the following commands, substituting the name of a Client Access server for "servername":

For Basic authentication:

**Set-PopSettings -Server "servername" -LoginType PlainTextLogin**

For Windows Integrated authentication

**Set-PopSettings -Server "servername" -LoginType PlainTextAuthentication**

2. Restart the POP3 service on that Client Access server.
3. Repeat for each of the Client Access servers to which you will be sending POP3 connections.

To load balance the POP Service, you will need to create a new farm, add servers to the farm, create a new L4 policy for TLS port **995** to backend server port **110** and configure health monitoring

1. Create a farm named "**CAS\_POP**" in **AppDirector -> Farms -> Farm Table** with these parameters:

- Farm Name = CAS\_POP
- Aging Time = 30
- Session mode = EntryPerSession
- Dispatch Method = Cyclic
- Connectivity checks = No Checks
- Leave all other fields as default

**Note:** No Persistency required for POP.

2. Create a server named "POP\_Svr1" and attach it to the farm "CAS\_POP" in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:

- Farm Name = CAS\_POP
- Server Address = 10.106.2.52
- Server Name = POP\_Svr1
- Server Port = 110
- Client NAT = Enable
- Client NAT Address Range = 10.1.9.1
- Leave all other fields as default

3. Create a server named "POP\_Svr2" and attach it to the farm "CAS\_POP" in **AppDirector -> Servers -> Application Servers -> Table** with these parameters:

- Farm Name = CAS\_POP
  - Server Address = 10.106.2.53
  - Server Name = POP\_Svr2
  - Server Port = 110
  - Client NAT = Enable
  - Client NAT Address Range = 10.1.9.1
  - Leave all other fields as default
4. Create an SSL policy in **AppDirector -> L4 Traffic Redirection -> SSL Policy** with these parameters:
- Policy name = Exchange\_POP\_SSL\_policy
  - Certificate = radware
  - Listening Server Port = 110 (servers listen on this port)
  - Backend SSL State = Disabled
  - Leave all other fields as default
5. Create a Layer 4 policy to select POP Service over TLS port 995 in **AppDirector -> Layer 4 Traffic Redirection -> Layer 4 Policies** with these parameters:
- Virtual IP = 10.1.3.152
  - L4 Port = 995
  - L4 Policy Name = POP\_TLS
  - Application = Genaric-SSL
  - SSL Policy = Exchange\_POP\_SSL\_policy
  - Application = Any
  - Farm Name = CAS\_POP
  - Leave all other fields as default
6. Create a check POP\_Svr1 server IP address 10.106.2.52 in **Health Monitoring -> Check Table**:
- Check name = POP\_Svr1
  - Method = POP3
  - Dest IP = 10.106.2.52
  - Dest Port = 110
  - Interval = 91
  - Timeout = 30
  - Arguments
    - Username = <the configured user name>
    - Password = <the configured password>
  - Leave all other fields as default
7. Create a check for POP\_Svr2 server IP address 10.106.2.53 in **Health Monitoring -> Check Table**:

- Check name = POP\_Svr2
  - Method = POP3
  - Dest IP = 10.106.2.53
  - Dest Port = 110
  - Interval = 91
  - Timeout = 30
  - Arguments
    - Username = <the configured user name>
    - Password = <the configured password>
  - Leave all other fields as default
8. Bind the check POP\_Svr1 to Farm CAS\_POP - 10.106.2.52 - 110 in **Health Monitoring -> Binding Table**.
9. Bind the check POP\_Svr2 to Farm CAS\_POP - 10.106.2.53 - 110 in **Health Monitoring -> Binding Table**.

## Appendix 6 – Backend Encryption

When encryption of traffic to the server side is required, the SSL Policy Configuration should be changed as follows:

### SSL Policy Configuration

**Note:** Configures in the SSL policy a reference is made to the “radware” pre-configured certificate, but you can import a certificate or create a new certificate in AppDirector. For more information on exporting, importing, or creating a certificate, see the *AppDirector User Guide*.

Create an SSL policy in **AppDirector -> L4 Traffic Redirection -> SSL Policy** with these parameters:

- Policy name = Exchange\_policy
- Certificate = radware
- Listening Server Port = 443
- Backend SSL State = Enabled
- Leave all other fields as default

## **Technical Support**

Radware offers technical support for all of its products through the Radware Certainty Support Program. Please refer to your Certainty Support contract, or the Radware Certainty Support Guide available at:

<http://www.radware.com/content/support/supportprogram/default.asp>.

For more information, please contact your Radware Sales representative or:

U.S. and Americas: (866) 234-5763

International: +972(3) 766-8666

© 2008 Radware, Ltd. All Rights Reserved. Radware and all other Radware product and service names are registered trademarks or trademarks of Radware in the U.S. and other countries. All other trademarks and names are the property of their respective owners.