



MS Exchange Server 2010: Highly Available, High Performing And Scalable Deployment With Coyote Point Equalizer



Prepared By: Mark Hoffmann
Coyote Point Systems Inc.

Copyright © 2011 Coyote Point Systems Inc.

All Rights Reserved.

The following are Trademarks or Registered Trademarks of Coyote Point Systems Incorporated in the United States and other countries:

Coyote Point™

Equalizer®

Equalizer VLB™

Envoy®

E250GX™

E350GX™

E450GX™

E650GX™

Microsoft and Outlook are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All other brand or product names used in this document are trademarks or registered trademarks of their respective companies or organizations.

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

Document Version: 1.4

July 2011

EXCHANGE 2010 AND APPLICATION DELIVERY.....	4
THE EQUALIZER DIFFERENCE	5
USING EQUALIZER WITH EXCHANGE 2010	6
HARDWARE AND SOFTWARE USED IN THIS GUIDE	7
CLUSTER CONFIGURATION SUMMARY	8
SERVER HEALTH CHECKS	8
LOAD BALANCING POLICY	8
SOURCE NETWORK ADDRESS TRANSLATION (SPOOF)	10
SSL OFFLOAD AND ACCELERATION	11
DATA COMPRESSION	11
THE EQUALIZER ADMINISTRATIVE INTERFACE	11
CONFIGURING OWA, OUTLOOK ANYWHERE, AND ACTIVESYNC	13
CREATING AN HTTPS CLUSTER FOR OWA / OA / AS	14
CREATE AN HTTP REDIRECT CLUSTER	15
CONFIGURING EQUALIZER FOR POP3	17
CREATING A POP3 LAYER 4 TCP CLUSTER	17
CONFIGURING EQUALIZER FOR IMAP4	18
CREATING AN IMAP4 LAYER 4 TCP CLUSTER	18
CONFIGURING EQUALIZER FOR RPC CLIENT ACCESS	19
CREATING THE RPC CA PORTMAPPER LAYER 4 TCP CLUSTER	19
CREATING THE RPC CA MAPI LAYER 4 TCP CLUSTER	20
CREATING THE RPC CA ADDRESS BOOK LAYER 4 TCP CLUSTER.....	21
CONFIGURING EQUALIZER FOR SMTP	23
CREATING AN SMTP LAYER 4 TCP CLUSTER	23
ENABLING SSL OFFLOADING IN EXCHANGE	24
ENABLING OUTLOOK WEB APP SSL OFFLOADING IN THE REGISTRY	24
CONFIGURE OUTLOOK ANYWHERE AND SSL OFFLOADING	24
ENABLING SSL OFFLOAD WHEN OUTLOOK ANYWHERE IS ALREADY CONFIGURED	25
ENABLING SSL OFFLOADING IN IIS	25
SUMMARY	27
ABOUT COYOTE POINT.....	27

Exchange 2010 and Application Delivery

Microsoft® Exchange Server 2010 was released in late 2009 as the successor to Microsoft Exchange Server 2007. It introduced a number of new features as well as changes to existing features. Enhancements were added with Service Pack 1 (SP1), released in August of 2010. This guide was produced using SP1.

Microsoft recognizes the need for load balancing client access in all but the smallest Exchange deployments. For Microsoft's overview of load balancing recommendations in Exchange 2010, please see:

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

As stated in the above document, client access in Exchange 2010 is concentrated at the Client Access Server (CAS) or middle tier of the Exchange Architecture. Placing a load balancer in front of the CAS array ensures that resources are used efficiently to provide the best user experience for both internal and external client access:

- “In earlier versions of Exchange, Outlook® connected directly to the Mailbox server hosting the user's mailbox, and directory connections were either proxied through the Mailbox server role or referred directly to a particular Active Directory® global catalog server. Now that these connections are handled by the Client Access server role, both external and internal Outlook connections must be load balanced across the array of Client Access servers in a deployment to achieve fault tolerance.

A load-balanced array of Client Access servers is recommended for each Active Directory site and for each version of Exchange.”

While software load balancers and reverse proxy solutions can be adequate for smaller deployments, larger deployments will benefit from the features and capacity provided by a hardware load balancer. Among other issues, Microsoft recognizes the following limitations with Windows Network Load Balancing, the most popular software based load balancing solution for Exchange:

- “Due to performance issues, we don't recommend putting more than eight Client Access servers in an array that's load balanced by WNLB.”
- “WNLB doesn't detect service outages. WNLB only detects server outages by IP address. This means if a particular Web service, such as Outlook Web App, fails, but the server is still functioning, WNLB won't detect the failure and will still route requests to that Client Access server. Manual intervention is required to remove the Client Access server experiencing the outage from the load balancing pool.”
- “WNLB configuration can result in port flooding, which can overwhelm networks.”
- “If you have more than eight Client Access servers in a single Active Directory site, your organization will need a more robust load balancing solution. Although there are robust

software load balancing solutions available, a hardware load balancing solution provides the most capacity.”

Another reason to deploy a hardware load balancer with Exchange 2010 is that Exchange uses a concept called a Database Availability Group (DAG) to provide high availability at the database level. A DAG is a group of up to 16 Mailbox servers that host a set of databases and provide automatic database-level recovery from failures that affect individual servers or databases. This architecture *requires* the use of an external load balancer to provide high availability above the database level.

The Equalizer Difference

There are a number of hardware load balancing products available on the market with a wide range of features and capabilities. Coyote Point’s Equalizer Application Delivery Controller differentiates itself by providing superior value; advanced acceleration features, high performance, and reliability born of over 10 years of industry experience.

Equalizer not only load balances Internet service requests across multiple servers, but also accelerates application performance and provides application aware features that monitor server load and improve server response times – by as much as 25%. In addition to basic load balancing, Equalizer provides:

- Automatic server and application health monitoring
- Intelligent, application aware load balancing policies (adaptive, least connections, fastest response time, static weight, server agent, custom, and round robin)
- Content switching – the ability to change load balancing behaviour based on the content of a client request
- Smart Events – the ability to specify administrative actions based on observed behaviour and conditions
- SSL offloading and acceleration
- Real time graphical performance monitoring and reporting
- Redundant High Availability (HA) configurations
- Tight integration with the VMware Infrastructure to provide higher application performance in a virtualized server environment
- HTTP Compression to reduce bandwidth requirements

For more information on how Equalizer can make your applications work better, faster, and more economically, please visit www.coyotepoint.com.

Using Equalizer with Exchange 2010

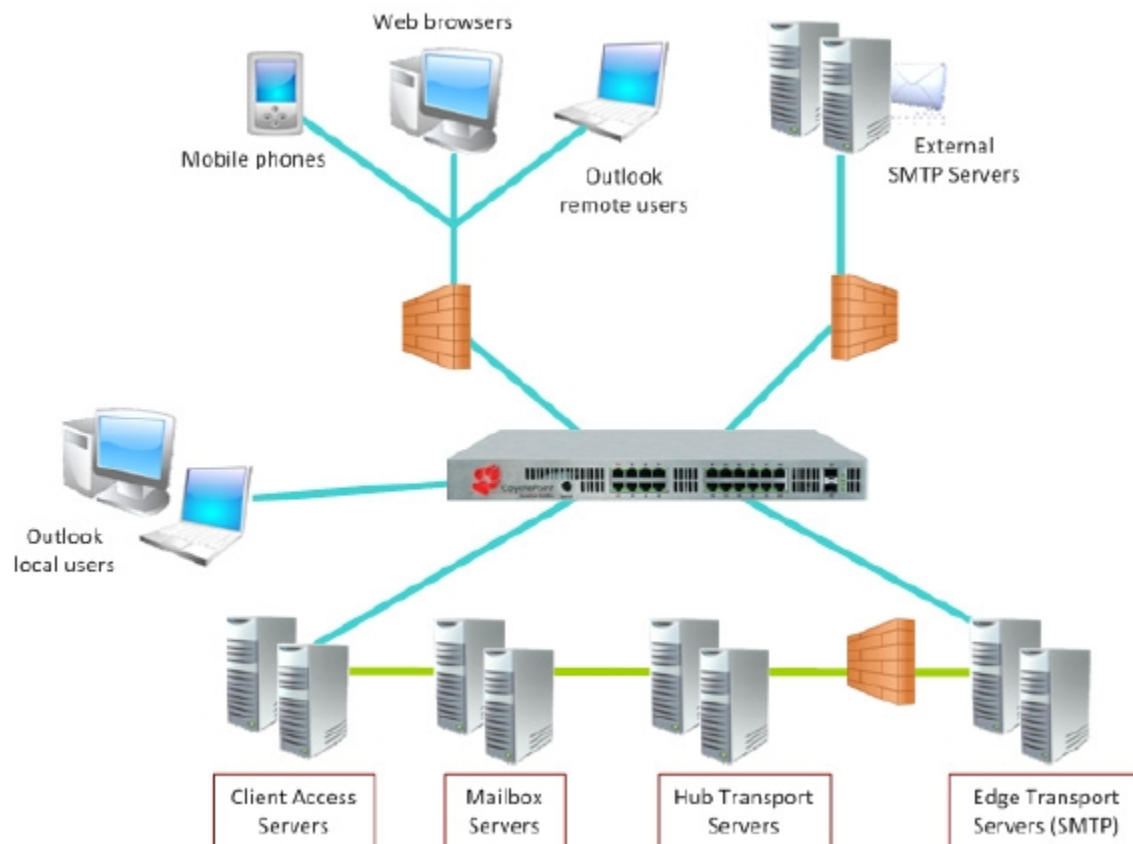
Setting up and configuring Microsoft Exchange 2010 requires significant planning to ensure adequate resources for deployment. Before beginning any deployment of Exchange 2010, thoroughly read and follow Microsoft's Exchange planning guide:

<http://technet.microsoft.com/en-us/library/aa998636.aspx>

In particular, it is vital that your hardware and network configuration has the processing capacity (CPU speed and memory), throughput, and bandwidth required for the number of users and client access methods that you need to deploy. Even in a test environment, you must ensure adequate resources, as described in the planning guide, so that your configuration functions efficiently.

For the purposes of this deployment guide, we assume a working Exchange deployment that will be augmented by the addition of Equalizer (or a pair of Equalizers in a failover configuration). If you are setting up a new deployment of Exchange, we recommend that you first set up your Exchange configuration without Equalizer, verify each of your intended client access methods, and then follow this document to deploy Equalizer into that configuration.

Logically, Equalizer sits in between clients accessing Exchange and the Exchange servers, as shown in the following diagram:



Clients can access Exchange via a number of applications and protocols:

Outlook Web App (OWA) – (known as Outlook Web Access in previous releases). Internal and external clients initiate OWA sessions over HTTP using a web browser, or Outlook Web App Light.

Outlook Anywhere (OA) – Outlook clients access Exchange by tunnelling the Outlook MAPI (Messaging Application Programming Interface) protocol over an HTTP connection.

Exchange ActiveSync (AS) – Mobile clients can synchronize with Exchange services, which push data to the mobile device, over an HTTP connection.

POP3 and IMAP4 – External and internal third-party mail programs use these protocols (Post Office Protocol v3 and Internet Message Access protocol v4) to retrieve and send email.

SMTP – External mail servers forward mail to Exchange through Edge Servers or Hub Transport Servers using the Simple Mail Transfer Protocol (SMTP).

RPC Client Access (RPC CA) – Connects MAPI clients (like Outlook) to the Client Access Server, rather than directly to the Mailbox server as in previous releases of Exchange.

All of the protocols above are routed through Equalizer and load balanced to the appropriate Client Access Server (or, in the case of SMTP, to an Edge Server deployed outside of the Exchange domain). Each of these applications/protocols requires a slightly different Equalizer configuration, as described in the remainder of this document.

Hardware and Software Used in This Guide

To develop this deployment guide, the following hardware and software was used:

- Equalizer model E650GX (with HTTP Compression and SSL Acceleration) EQ/OS Version 8.6
- Custom Server hardware running VMware ESX 4
- Several VM servers running Microsoft Server 2008 R2 (64-bit)
- Microsoft Exchange Server 2010 SP1
- Appropriately configured clients to test client access

Note that the hardware and software required for your configuration will vary from the above depending on your testing and production environment. Microsoft Hyper-V, for example, could be used in place of VMware.

If you do not have locally available clients of all types, Microsoft offers two alternatives that you can use in a testing environment to validate your configuration prior to putting it into production:

- The [Exchange Load Generator 2010](#) can be installed on a local server and can be configured to generate Exchange traffic for the various protocols.

- The [Exchange Remote Connectivity Analyzer](#) is an online tool that you configure to test your Internet accessible Exchange configuration.

Click on the links above for more details on configuring and using these tools.

Cluster Configuration Summary

The following table shows the basic configuration required for each application/protocol:

Application/Protocol	Cluster Type	Cluster Port	Server Port	Affinity (Persistence)	Server Health
OWA	L7 HTTPS	443	80	Equalizer Cookie	TCP (port 80)
Outlook Anywhere	L7 HTTPS	443	80	Equalizer Cookie	TCP (port 80)
ActiveSync	L7 HTTPS	443	80	Equalizer Cookie	TCP (port 80)
POP3	L4 TCP	995	995	None	TCP (port 995)
IMAP4	L4 TCP	993	993	None	TCP (port 993)
SMTP	L4 TCP	25	25	None	TCP (port 25)
RPC CA (Portmapper)	L4 TCP	135	135	Client IP (sticky)	TCP (port 135)
RPC CA (MAPI)	L4 TCP	59532	59532	Client IP (sticky)	TCP (port 59532)
RPC CA (AB)	L4 TCP	59533	59533	Client IP (sticky)	TCP (port 59533)

Server Health Checks

By default, Equalizer probes server health using ICMP and TCP probes. You can also enable Active Content Verification (verifies server availability via specific content) or server agents (user-supplied programs running on the server), if more specific probing is desired. If your servers are Virtual Machines running on VMware, you can enable Equalizer VLB agents -- which return detailed server health information from VMware that can be used for load balancing policy decisions as well as health checks. For more information, see the *Equalizer Installation and Administration Guide*.

Load Balancing Policy

In previous versions of Microsoft Exchange, Microsoft recommended using the **least connections** load balancing policy, which routes requests to servers that are more lightly loaded in terms of number of open connections. Microsoft has since changed its

recommendation, since it is possible that using a load balancing algorithm like **least connections** can lead to overloading a server when it is first brought online. Microsoft now recommends using a policy that does not depend on weighted criteria, such as **round robin** – which simply routes requests evenly across all available servers, regardless of performance. The result is that while **round robin** makes it less likely that any one server will be overloaded when it is brought online, it leads to an imbalance in the distribution of requests across all servers when servers are brought offline and online.

It should be noted that Equalizer's **least connections** setting is less prone to overloading a new server, since Equalizer tries to avoid overloading the server by also checking the server's response time and server agent value (if used). In this way, the **least connections** policy attempts to optimize the balance of connections to servers in the cluster. Other policies, such as **adaptive** and **fastest response**, behave similarly.

In addition, the speed with which all load balancing policies adjust load in response to changes in server characteristics is controlled by the **responsiveness** setting, which is set to **medium** by default. Setting **responsiveness** appropriately when using weighted load balancing policies can help reduce the potential for overloads.

Besides **round robin** and **least connections**, Equalizer also offers these load balancing policies:

static weight -- distributes requests among the servers depending on their assigned initial weights. A server with a higher initial weight gets a higher percentage of the incoming requests. Think of this method as a weighted round robin implementation. Static weight load balancing does not support Equalizer's adaptive load balancing feature – that is, Equalizer does not dynamically adjust server weights based on server performance.

adaptive -- distributes the load according to these performance indicators for each server.

- **Server response time** is the length of time for the server to begin sending reply packets after Equalizer sends a request.
- **Active connection count** is the number of connections currently active on the server.
- **Server agent value** is the value returned by the server agent daemon (if any) running on the server.

fastest response -- dispatches the highest percentage of requests to the server with the shortest response time. Equalizer does this carefully: if Equalizer sends too many requests to a server, the result can be an overloaded server with slower response time. The fastest response policy optimizes the cluster-wide response time. The fastest response policy also checks the number of active connections and server agent values (if configured); but both of these have less of an influence than they do under the adaptive load balancing policy. For example, if a server's active connection count and server agent values are high, Equalizer might not dispatch new requests to that server even if that server's response time is the fastest in the cluster.

server agent -- dispatches the highest percentage of requests to the server with the lowest server agent value. In a similar way to Fastest Response, Equalizer tries to avoid overloading the

server by checking the number of connections and response time. This method only works if server agents are running on all servers in the cluster.

custom – allows you to set the relative importance of load balancing criteria to specifically configure a policy that meets your needs.

Source Network Address Translation (spoof) Setting

The cluster **spoof** option controls which IP address is used as the source address in packets sent to the servers behind Equalizer; this is more generally known as ‘Source Network Address Translation’ or SNAT:

- When the **spoof** option is enabled, no SNAT is performed on client requests – that is, servers will see the client’s IP address as the source IP in all packets received from Equalizer. *This is the default setting for a new cluster.*
- Disabling the **spoof** option turns on SNAT – Equalizer translates the client IP address to Equalizer’s IP address on the VLAN/subnet. So, the server receiving the request will see Equalizer’s IP address as the source IP address.

In general, Microsoft recommends using SNAT (*disabling* the **spoof** option) for all configurations, although it may not be appropriate in circumstances where it is desirable to see the real client IP address at the server (e.g., for logging):

- In a multiple VLAN/subnet configuration, you can either:
 - Enable the **spoof** option and set the default gateway on each server to Equalizer’s IP address on the same subnet/VLAN (or, use static routes to send responses to Equalizer’s IP address).

Or:

 - Use SNAT (disable **spoof**).
- In a single VLAN/subnet configuration, SNAT should be used. The **spoof** option should be *disabled*, so that servers see Equalizer’s IP address in the request and send responses back to Equalizer.

Note that Direct Server Return (DSR) configurations are supported by Exchange, but in general SNAT-enabled configurations are recommended by Microsoft to avoid the additional complexity and drawbacks of DSR. For example: to use DSR with any load balancer requires configuration of a special loopback adapter on each server and is supported only for Layer 4 services.

See the Equalizer *Installation and Administration* Guide for more information on Equalizer network configuration.

SSL Offload and Acceleration

SSL offload is performed by Equalizer for Layer 7 HTTPS clusters. The instructions in this document show you how to upload a server certificate to an HTTPS cluster on Equalizer, as well as perform the necessary operations for each offloaded protocol on Exchange 2010.

SSL Acceleration is provided on E450GX and E650GX model Equalizers via special hardware available only on those models.

Data Compression

Data compression is available on the E650GX model Equalizer only. Compression is enabled on a cluster by cluster basis by turning on the compress check box in the cluster's Networking configuration. Data compression on the E650GX provides throughput in most configurations that is 3 to 5 times the throughput observed when compression is not used.

The Equalizer Administrative Interface

This guide assumes that you have already set up Equalizer on your organization's network. Full instructions are available in the printed startup guides and CD-ROM delivered with your Equalizer. Documentation is also available from <http://docs.coyotepoint.com/>). Once Equalizer has an IP address on the network, open the Administrative Interface by opening the following URL in your web browser:

```
http://<Equalizer_IP_addr>
```

Where <Equalizer_IP_addr> is Equalizer's management IP address. Log in to Equalizer using a login with administrator privileges. This opens the graphical user interface, as shown in the following figure:

Equalizer System Information

current user	touch
user permissions	administrator
Equalizer version	8.6.0c
system ID	[blurred]
serial no.	[blurred]
platform	e650gx Rev. 1.0
system name	eq_172.16.0.140

Copyright © 1998-2010 Coyote Point Systems Inc. All Rights Reserved.
Coyote Point, Equalizer, the Equalizer logo, Envoy, and the Envoy logo are trademarks of Coyote Point Systems Incorporated.

The clusters, servers, responders, and match rules you create for Exchange 2010 will be displayed in the left frame, while configuration details are displayed and modified in the right frame:

- Click an object in the left frame to display the configuration details for that object.
- Right-click an object in the left frame to display commands for that object.
- Click **Equalizer** to display global configuration parameters.
- Click **Help > Context Help** at any time to display documentation for the currently displayed configuration details.

Configuring OWA, Outlook Anywhere, and ActiveSync

Equalizer is easily configured for Outlook Web App (OWA), Outlook Anywhere (OA), and ActiveSync (AS) using an HTTPS cluster. This offloads SSL processing from the MS Exchange 2010 Client Access Server (CAS) array behind Equalizer.

In the descriptions below, <FQDN> means the fully qualified domain name of the Exchange CAS array, e.g., mail.example.com. This FQDN must be changed in DNS and/or Active Directory to point to the Equalizer HTTPS cluster IP.

- Outlook Web App connections originate from client web browsers and use the URL `https://<FQDN>/rpc` to access Exchange.
- Outlook Anywhere (known in previous versions of Exchange as RPC over HTTP) connections originate from Outlook 2010, Outlook 2007, and Outlook 2003 clients, and use the URL `https://<FQDN>/ExchWeb` to connect to Exchange servers.
- Exchange ActiveSync connections can originate from any EAS enabled mobile device and use the URL `https://<FQDN>/Microsoft-Server-ActiveSync` to access Exchange.

We'll also show you how to create an additional HTTP cluster and a responder to redirect client requests that mistakenly specify the HTTP protocol instead of HTTPS.

See the section "Enabling SSL Offloading in Exchange" for how to enable SSL offloading for these applications/protocols on your Exchange CAS servers.

[Note that any or all of these services can also be configured on Equalizer using a Layer 4 cluster, but doing so has several consequences: all SSL processing will be performed on the server (since SSL offload is not available with Layer 4 clusters), persistence will be provided by IP address (instead of cookies), and you may need a separate SSL certificate for any services provided via separate clusters. We do not recommend using Layer 4 clusters unless you need to support clients that do not work with cookie persistence. The procedures in this document assume that a single Layer 7 HTTPS cluster will be used to provide client access to the OWA, OA, and AS Exchange services.]

Creating an HTTPS Cluster for OWA / OA / AS

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select Layer 7 HTTPS and click the Next button (>).
3. Do the following:
 - a. Enter a name for the cluster (e.g., "OWA_OA_HTTPS"), or accept the default.
 - b. Enter the **IP address** for the cluster. The Fully Qualified Domain Name that clients use to connect to OWA and OA must resolve to this IP address instead of going through the CAS array directly.
 - c. Enter the cluster **port**; accept the default of **443**.
 - d. Click the **Next** button (>).
4. Click commit to create the cluster; it appears in the left pane and the Configuration tab for the cluster opens in the right pane.
5. Open the **Networking** tab and increase the client timeout value to 20 seconds. This is the amount of time that Equalizer waits for the client to send all the headers in a client request. If this timer expires, Equalizer sends a timeout to the client. Twenty seconds is the recommended value for client timeout in HTTPS clusters.

Click **commit**.

6. Open the **LB Algorithm** tab and select **round robin** from the policy drop-down box. Then click **commit**.
7. Open the **Security > SSL** tab for the cluster to supply the SSL certificate for the cluster. The certificate must be one of the following:
 - a wildcard certificate
 - a Universal Communications Certificate (UCC),
 - a certificate that specifically matches the URL that clients will use to connect to OWA and OA.

Use the **Browse** button to locate the certificate on your local system, then click **upload** to add it to the cluster.

8. If your configuration requires a client certificate for the cluster, click the **client** button and repeat the previous step to upload the client certificate to Equalizer.
9. Right click on the new cluster name in the left frame and select **Add Server** from the popup menu. Do the following:
 - a. Enter an **IP address** and **port** for one of the CAS servers configured for OWA behind Equalizer.

- b. Specify port **80** unless your IIS server on the CAS is configured to use another port.
 - c. Uncheck the **quiesce on creation** option.
 - d. Click **Next (>)** to confirm your settings, or press the **Enter** key to create the server now.
10. Repeat the previous step for each server in your CAS array that is configured for OWA, OA, and OAS.

Create an HTTP Redirect Cluster

Sometimes a client user will specify `http://` instead of `https://` when attempting to connect to OWA, OA, or AS. This section shows you how to create an Equalizer HTTP cluster and responder that will automatically redirect such requests to the HTTPS cluster created in the previous section.

First we create the cluster, then create the responder, and then associate the responder with the cluster via a new cluster match rule.

[Note that no servers are added to this cluster, since all requests will be redirected to the HTTPS cluster.]

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select **Layer 7 HTTP** and click the > button.
3. Enter the following information:
 - a. A **name** for the cluster (e.g., "OWA_OA_HTTP"), or accept the default.
 - b. The **IP address** for the cluster. This is the same IP address that you used to create the HTTPS cluster in the previous section.
 - c. The cluster **port**; accept the default of 80.
 - d. Click the > button.
4. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
5. Right-click Responder in the left frame object tree and select Add New Responder.
6. In the dialog that appears, do the following:
 - a. Enter a **name** for the responder, such as **Exch_http_redirect**.
 - b. Ensure that the **Redirect** button is selected.
 - c. Select an appropriate **Status** code from the drop-down box.
 - d. Type the following in the URL box:
`https://$1`

- e. Type the following in the Regex box:
`http://(.*)`
- f. Click **commit** to create the responder.
7. In the left frame, right-click on the name of the cluster created above (see Step 3) and select **Add Match Rule** from the popup menu. Enter a name for the match rule (or accept the default) and click **commit**. The new match rule is added to the left frame and the **Configuration** tab for the match rule opens in the right frame.
8. On the match rule **Configuration** tab, select the name of the responder you created in Step 6 above from the **responder** drop-down box near the bottom of the tab. Click **commit**.

Configuring Equalizer for POP3

To support mailbox access from POP3 clients you need to start the POP3 service on the Exchange Client Access Servers. For clients to send email through Exchange, you'll also need to configure SMTP. POP3 is load balanced using a Layer 4 TCP cluster, with no persistence. SSL offloading is not performed by Equalizer for POP3.

Creating a POP3 Layer 4 TCP Cluster

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select **Layer 4 TCP** and click the **Next** button (>).
3. Enter the following information:
 - a. A name for the cluster (e.g., "POP3_TCP"), or accept the default.
 - b. The IP address for the cluster. The Fully Qualified Domain Name that clients use to connect to POP3 must resolve to this IP address instead of going through the CAS array directly.
 - c. The cluster port; enter the standard POP3 over SSL port, 995.
 - d. Click the **Next** button (>).
4. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
5. Open the LB Algorithm tab and select **round robin** from the policy drop-down box. Click **commit**.
6. Right click on the cluster name in the left frame and select Add Server from the popup menu. Do the following:
 - a. Enter an IP address and port for one of the CAS servers configured for OWA behind Equalizer. Specify port 995.
 - b. Uncheck the **quiesce on creation** option.
 - c. Click **Next** (>) to confirm your settings, or press the **Enter** key to create the server now.
7. Repeat the previous step for each server in your CAS array that is configured for POP3.

Configuring Equalizer for IMAP4

To support mailbox access from IMAP4 clients, you need to start the IMAP4 service on the Exchange Client Access Servers. For clients to send email through Exchange, you'll also need to configure SMTP.

IMAP4 is load balanced using a Layer 4 TCP cluster, and persistence is not needed for IMAP4. SSL offloading is not performed by Equalizer for IMAP4.

Creating an IMAP4 Layer 4 TCP Cluster

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select **Layer 4 TCP** and click the **Next** button (>).
3. Enter the following information:
 - a. A name for the cluster (e.g., "IMAP4_TCP"), or accept the default.
 - b. The IP address for the cluster. The Fully Qualified Domain Name that clients use to connect to IMAP4 must resolve to this IP address instead of going through the CAS array directly.
 - c. The cluster port: enter the standard IMAP4 over SSL port, 993.
 - d. Click the **Next** button (>).
4. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
5. Open the **LB Algorithm** tab and select **round robin** from the policy drop-down box. Click **commit**.
6. Right click on the cluster name in the left frame and select **Add Server** from the popup menu. Do the following:
 - a. Enter an **IP address** and **port** for one of the CAS servers configured for OWA behind Equalizer. Specify port **993**.
 - b. Uncheck the **quiesce on creation** option.
Click **Next (>)** to confirm your settings, or press the **Enter** key to create the server now.
7. Repeat the previous step for each server in your CAS array that is configured for IMAP4.

Configuring Equalizer for RPC Client Access

Load balancing RPC Client Access services through Equalizer requires three Layer 4 clusters, one for the Portmapper, one for MAPI, and one for the Address Book (AB) service. The instructions below assume that you are using a static port configuration for RPC Client Access as recommended by Microsoft.

Creating the RPC CA Portmapper Layer 4 TCP Cluster

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select **Layer 4 TCP** and click the **Next** button (>).
3. Enter the following information:
 - a. A name for the cluster (e.g., "Portmap_TCP"), or accept the default.
 - b. The **IP address** for the cluster. The Fully Qualified Domain Name that clients use to connect for RPC Client Access must resolve to this IP address instead of going through the CAS array directly.
 - c. The cluster port; enter **135** for **start port** and leave the end port blank.
 - d. Click the **Next** button (>).
4. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
5. Disable the cluster **spoof** option and click **commit**.
6. Open the **LB Algorithm** tab and select **round robin** from the policy drop-down box. Click **commit**.
7. Click on the **Persistence** tab in the right frame and do the following:
 - a. Enable the **inter-cluster sticky** option.
 - b. Enter a **sticky time** equal to the number of seconds that client/server connections will persist. Microsoft documentation states that Outlook clients assume that all RPC connections are made to the same server, and Outlook opens multiple connections per session, so you want to set this value to the longest anticipated user session, so that the user does not have to close their current connection and reconnect during a "session".

Microsoft generally recommends one hour (3600 seconds) for load balancer timeouts, though some application may require longer timeouts.
 - c. Click **commit**.
8. Right click on the cluster name in the left frame and select **Add Server** from the popup menu. Do the following:

- a. Enter an **IP address** for one of the CAS servers configured for RPC CA behind Equalizer. Accept the default **port** displayed -- this sets the same port on the servers as we set on the cluster in Step 3. TCP Health check probes will also use this port.
 - b. Uncheck the **quiesce on creation** option.
 - c. Click **Next (>)** to confirm your settings and then **commit**; or, press the **Enter** key to create the server now.
9. Repeat the previous step for each server in your CAS array that is configured with the RPC CA Portmapper.

Creating the RPC CA MAPI Layer 4 TCP Cluster

1. On all MAPI servers, use the Windows **regedit** tool to set the static port for the MAPI protocol by setting the following registry key, as in this example, which sets the port to 59532:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\MSExchangeRPC\ParametersSystem]
"TCP/IP Port"=dword:0000e88c
```

2. Right-click Equalizer in the left frame and select the Add Cluster command from the popup menu.
3. Select Layer 4 TCP and click the Next button (>).
4. Enter:
 - a. A name for the cluster (e.g., "MAPI_TCP"), or accept the default.
 - b. The IP address for the cluster. The Fully Qualified Domain Name that clients use to connect to RPC Client Access services must resolve to this IP address instead of going through the CAS array directly.
 - c. The cluster port; enter the port configured for MAPI on the RPC CA server (e.g., 59532) for start port and leave the end port blank.
 - d. Click the **Next** button (>).
5. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
6. Disable the cluster **spoof** option and click **commit**.
7. Open the **LB Algorithm** tab and select **round robin** from the policy drop-down box. Click **commit**.
8. Click on the **Persistence** tab in the right frame and do the following:
 - a. Enable the **inter-cluster sticky** option.
 - b. Enter a **sticky time** equal to the number of seconds that client/server connections will persist. Microsoft documentation states that Outlook clients assume that all RPC

connections are made to the same server, and Outlook opens multiple connections per session, so you want to set this value to the longest anticipated user session, so that the user does not have to close their current connection and reconnect during a “session”.

Microsoft generally recommends one hour (3600 seconds) for load balancer timeouts, though some application may require longer timeouts.

- c. Click **commit**.
9. Right click on the cluster name in the left frame and select **Add Server** from the popup menu. Do the following:
 - a. Enter an IP address and port for one of the CAS servers configured for RPC CA behind Equalizer. Accept the default port displayed -- this sets the same port on the servers as we set on the cluster in Step 3. TCP Health check probes will also use this port.
 - b. Uncheck the quiesce on creation option.
 - c. Click Next (>) to confirm your settings, or press <Enter> to create the server now.
10. Repeat the previous step for each server in your CAS array that is configured with MAPI services.

Creating the RPC CA Address Book Layer 4 TCP Cluster

Note: In Step 1, below, the Address Book static port is set using the method appropriate for Microsoft Exchange 2010 SP1. If you have an earlier version of Exchange running on a server, please see the Microsoft 2010 documentation for the version you are running for how to set the Address Book static port.

1. On all Address Book servers, use regedit to set the static port for the Address Book service by navigating to the Registry location shown below and setting the “RpcTcpPort” key as shown in this example (which sets the port to 59533):

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\MSEExchangeAB\Parameters]
"RpcTcpPort"="59533"
```

2. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
3. Select **Layer 4 TCP** and click the **Next** button (>).
4. Enter:
 - a. A name for the cluster (e.g., “AddrBook_TCP”), or accept the default.
 - b. The IP address for the cluster. The Fully Qualified Domain Name that clients use to connect to RPC Client Access services must resolve to this IP address instead of the CAS array IP address.
 - c. The cluster port; enter the port configured for MAPI on the RPC CA server (e.g., 59533) for start port and leave the end port blank.

- d. Click the **Next** button (>).
5. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
6. Disable the cluster **spoof** option and click **commit**.
7. Open the **LB Algorithm** tab and select **round robin** from the **policy** drop-down box. Click **commit**.
8. Click on the **Persistence** tab in the right frame and do the following:
 - a. Enable the **inter-cluster sticky** option.
 - b. Enter a **sticky time** equal to the number of seconds that client/server connections will persist. Microsoft documentation states that Outlook clients assume that all RPC connections are made to the same server, and Outlook opens multiple connections per session, so you want to set this value to the longest anticipated user session, so that the user does not have to close their current connection and reconnect during a “session”.

Microsoft generally recommends one hour (3600 seconds) for load balancer timeouts, though some application may require longer timeouts.
 - c. Click **commit**.
9. Right click on the cluster name in the left frame and select **Add Server** from the popup menu. Do the following:
 - a. Enter an **IP address** and **port** for one of the CAS servers configured for RPC CA behind Equalizer. Accept the default port displayed -- this sets the same port on the servers as we set on the cluster in Step 3. TCP Health check probes will also use this port.
 - b. Uncheck the **quiesce on creation** option.
 - c. Click **Next** (>) to confirm your settings, or press <Enter> to create the server now.
10. Repeat the previous step for each server in your CAS array that is configured with Address Book services.

Configuring Equalizer for SMTP

SMTP connections in Exchange 2010 may be configured using either Edge Transport Servers or Hub Transport Servers. Just use the appropriate IP addresses for your configuration when adding servers in Step 6, below.

Creating an SMTP Layer 4 TCP Cluster

1. Right-click **Equalizer** in the left frame and select **Add Cluster** from the popup menu.
2. Select **Layer 4 TCP** and click the **Next** button (>).
3. Do the following:
 - a. Enter a **name** for the cluster (e.g., "SMTP_TCP"), or accept the default.
 - b. Enter an **IP address** for the cluster. The Fully Qualified Domain Name that clients use to connect to SMTP must resolve to this IP address instead of going through the CAS array directly.
 - c. Enter the cluster **port**; enter the standard SMTP port, **25**.
 - d. Click the **Next** button (>).
4. Click **commit** to create the cluster; it appears in the left pane and the **Configuration** tab for the cluster opens in the right pane.
5. Open the **LB Algorithm** tab and select **round robin** from the **policy** drop-down box. Click **commit**.
6. Right click on the cluster name in the left frame and select **Add Server** from the popup menu. Do the following:
 - a. Enter an **IP address** and **port** for one of the Edge Transport or Hub Transport Servers configured for SMTP behind Equalizer. Specify port **25**.
 - b. Uncheck the **quiesce on creation** option.
 - c. Click **Next** (>) to confirm your settings, or press <Enter> to create the server now.
7. Repeat the previous step for each server in your CAS array that is configured for SMTP.

Enabling SSL Offloading in Exchange

SSL offloading means that the client SSL connection is terminated at Equalizer, and Equalizer communicates with the CAS using unencrypted HTTP. SSL offloading significantly improves CAS performance and simplifies certificate management, since all SSL certificates reside on Equalizer, and Equalizer performs all the CPU-intensive SSL processing.

Equalizer provides SSL offloading for Layer 7 HTTPS clusters, so in our deployment we will enable SSL offloading for the following Exchange services:

- Outlook Web App – you must enable SSL offloading in the registry and in IIS on each CAS
- Outlook Anywhere – enable SSL offloading in Outlook Anywhere properties and IIS on each CAS
- Exchange ActiveSync – enable SSL offloading in IIS on each CAS

Enabling Outlook Web App SSL Offloading in the Registry

Enabling SSL offload for Outlook Web App requires that you create a new key in the Windows Registry, as described below.

1. From the desktop, click **Start > Run**, and enter `regedit` into the text box. Click **OK**.
2. Use the left pane tree to navigate to the following registry location:
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchange OWA`
3. On the Registry Editor menu, click **Edit > New > DWORD**. A new key appears in the right pane of the editor.
4. Enter **SSLOffloaded** for the new key **Name** and press **<Enter>**.
5. Right-click the new **SSLOffloaded** key and select **Modify** from the popup menu.
6. In the **Value** data field, type **1** (the number one). Click **OK** to save your changes.
7. Select **File > Exit** to close the Registry Editor.
8. Go to the section “Enabling SSL Offloading in IIS”.

Configure Outlook Anywhere and SSL Offloading

Use this procedure to turn on Outlook Anywhere and enable SSL offloading at the same time. Do the following for each CAS server:

1. Launch the **Exchange Management Console** (EMC) from the **Start** menu.
2. In the console tree at left, navigate to **Server Configuration > Client Access**.

3. In the **Client Access** pane (middle pane), click the server on which you want to enable Outlook Anywhere.
4. In the **action** pane, click **Enable Outlook Anywhere**.
5. In the **Enable Outlook Anywhere** wizard, in the box under **External host name**, type the external host name for your organization.
6. Select an available external authentication method. You can select **Basic authentication** or **NTLM authentication**.
7. Enable the check box next to **Allow secure channel (SSL) offloading**.
8. Click **Enable** to apply these settings and enable Outlook Anywhere.
9. Click **Finish** to close the Enable Outlook Anywhere wizard.
10. Go to the section "Enabling SSL Offloading in IIS".

Enabling SSL Offload when Outlook Anywhere is Already Configured

Use this procedure when you have already configured Outlook Anywhere and want to modify it to enable SSL offloading. Do the following for each CAS server:

1. Launch the **Exchange Management Console (EMC)** from the **Start** menu.
2. In the console tree at left, navigate to **Server Configuration > Client Access**.
3. In the **Client Access** pane (middle pane), right-click the name of the server on which you want to enable SSL offloading for Outlook Anywhere, and select **Properties** from the popup menu.
4. In the **Properties** window, open the **Outlook Anywhere** tab.
5. Enable the check box next to **Allow secure channel (SSL) offloading**.
6. Click **OK** to apply these settings and enable Outlook Anywhere on the server.
7. Go to the section "Enabling SSL Offloading in IIS".

Enabling SSL Offloading in IIS

We now need to re-configure the IIS web site on each CAS, so that it will accept HTTP connections instead of requiring HTTPS connections. Do the following on each CAS server in the configuration.

1. From the desktop, click **Start > Administrative Tools > Internet Information Services (IIS) Manager**.

2. In the left pane tree, expand **Server_name > Sites > Default Web Site** to display all the default web site home pages.
3. Do the following for each of the components that you are routing through an Equalizer HTTPS cluster, as shown in this table:

Exchange Component	Home Page
Outlook Anywhere	Exchweb
Outlook Web Access	owa
ActiveSync	Microsoft-Server-ActiveSync

- a. Click the name of the home page in the left pane.
 - b. In the middle pane, open **SSL Settings** and disable the **Require SSL** check box.
 - c. In the **Actions** pane at right, click **Apply**.
4. When you are done, close the **IIS Manager**.

Summary

Equalizer provides the load balancing, application acceleration, and high availability features demanded by medium to large Microsoft Exchange Server 2010 configurations. This document has presented a step-by-step guide to configuring Equalizer's features for an Exchange 2010 environment.

If you want more information on how Coyote Point can add value to your deployment plans please email info@coyotepoint.com or call us at 1-877-367-2696.

About Coyote Point

Coyote Point has been an application delivery innovator for over 10 years. In 1999 we introduced our first server load balancer and we've shipped thousands of units since then. Today, Coyote Point leads the industry in producing reliable, high performance Application Delivery Controllers that can be scaled to meet any application delivery environment. Coyote Point's Equalizer ADC products are deployed by small, medium and large enterprises, including some of the busiest sites on the web. At Coyote Point, we pride ourselves on delivering value to our customers. Our products perform as advertised and are easy and enjoyable to use and deploy. With a versatile and powerful architecture, Coyote Point provides the highest value while enabling customers to optimize businesses that rely on Internet-based infrastructure.

Please visit our website at www.coyotepoint.com.