Stingray Traffic Manager Solution Guide

Load Balancing and Optimization for Microsoft Exchange 2010 Client Access Servers

Riverbed Technical Marketing

Version 2.7
Contents

1.0 Solution Overview ..................................................................................................................3
1.1 Riverbed Stingray™ Traffic Manager ..................................................................................3
1.2 Microsoft® Exchange 2010 Server ......................................................................................3
1.3 Microsoft Exchange 2010 High Availability .........................................................................3
2.0 Exchange 2010 architecture ..................................................................................................4
2.1 Single Client Access Server ..................................................................................................4
2.2 Client Access Server Array ...................................................................................................4
2.3 Client Access server changes ...............................................................................................5
3.0 Prerequisites and Configuration tips for Microsoft Exchange 2010 Client Access Role on CAS Array ..................................................................................................................5
3.1 Setting up Microsoft Exchange 2010 CAS Array ..................................................................5
3.2 RPC Client Access (MAPI) ...................................................................................................5
3.3 Exchange 2010 Address Book Service ...............................................................................6
3.4 Exchange 2010 Public Folder connections .........................................................................8
3.5 Changing the External URLs of Exchange HTTP services for respective Virtual Directories on CAS IIS Server .................................................................9
4.0 Stingray Traffic Manager Configuration: Separate Virtual Server For Each Microsoft Exchange 2010 CAS Client Access HTTP Service ....11
4.1 Overview ...............................................................................................................................11
4.2 Configuring Stingray Traffic Manager for Outlook Web App (OWA) ....................................11
4.3 Configuring Stingray Traffic Manager to Redirect All HTTP requests to SSL ..................19
4.4 Configuring Stingray Traffic Manager for Outlook Anywhere ..........................................22
4.5 Configuring Stingray Traffic Manager for ActiveSync .....................................................28
4.6 Configuring Stingray Traffic Manager for Auto Discover .................................................36
4.7 Configuring Stingray Traffic Manager for MAPI RPC Client Access .................................38
4.8 Configuring Stingray Traffic Manager for POP3 ..................................................................46
4.9 Configuring Stingray Traffic Manager for IMAP4 ............................................................49
5.0 Stingray Traffic Manager Configuration: Single Virtual Server for OWA, OA, ECP, EWS, Active-Sync and Auto-Discover using Traffic Script .................................................................................50
6.0 Configuration Summary Of All Microsoft Exchange 2010 Services on Traffic Manager ..........................................................................................................................55
7.0 Configuration Worksheet ....................................................................................................56
8.0 Extra Optional Functionality ...............................................................................................56
9.0 Implementation .....................................................................................................................57
9.1 Physical Network Deployment ............................................................................................57
9.2 Domain Name Service .........................................................................................................57
10.0 Monitoring ..........................................................................................................................57
11.0 CONCLUSION .....................................................................................................................57
APPENDIX ..................................................................................................................................57
1. Traffic Script Code for redirecting all HTTP requests to SSL referenced as “OWA_Redirect_SSL” in section 4.3 .................................................................57
3. Traffic Script code for ActiveSync Persistence referenced as “Exchange 2010 ActiveSync Persistence” in section 4.5 ..............58
Changes in Current Deployment Guide .......................................................................................60
1.0 Solution Overview

1.1 Riverbed Stingray™ Traffic Manager

Despite increasing traffic loads, rapid change, and complex deployment infrastructures, online applications are still expected to deliver consistently excellent service levels. Stingray traffic management solutions provide complete control over user traffic, allowing administrators to accelerate, optimize, and secure key business applications. Now it’s possible to deliver these services more quickly and ensure the best possible performance across any deployment platform.

Application delivery controllers accelerate transactions, maximize availability, manage security policies, and provide a point of control to monitor and manage application traffic. Stingray Traffic Manager is a software-based ADC that provides unprecedented scale and flexibility to deliver applications across the widest range of environments, from physical and virtual data centers to public and hybrid clouds.

Stingray Traffic Manager Benefits include:

- **Speed**: Accelerate services, increase capacity, and reduce costs by offloading performance-draining tasks such as SSL and compression onto Stingray Traffic Manager’s optimized implementations. Cache commonly requested content and optimize traffic delivery to applications so they’ll run as fast as they would in a perfect benchmark environment.
- **Reliability**: Improve application availability by intelligently distributing traffic, avoiding failed or degraded servers, monitoring performance problems, and shaping traffic spikes.
- **Improved security**: Stingray Traffic Manager operates as a deny-all gateway, only admitting traffic types it has been configured to admit. This provides full control over how traffic is internally routed. High-performance inspection can interrogate any part of a request or response to apply global filtering or scrubbing policies. The Stingray Application Firewall option also protects against a broad range of web application attacks.
- **Ease of management**: Stingray Traffic Manager makes it easy to manage how users interact with applications, and the infrastructure those applications depend on. Use it to shape, prioritize, and route traffic, to drain infrastructure resources prior to maintenance, and to upgrade user sessions across application instances, all while preserving the user experience that business demands.

1.2 Microsoft® Exchange 2010 Server

Built to deliver the enterprise-grade security and reliability that businesses require, Microsoft Exchange provides email, calendar and contacts on your PC, phone and web browser.

- Support for a variety of browsers, including Internet Explorer, Firefox, Safari and Chrome, allows you to work and collaborate no matter where you are.
- Mobile sync to hundreds of devices, including Windows Phone, iPhone, and Android, means you can access and update your info while on the go.
- Multi-layered anti-spam filtering with continuous updates helps guard against spam and phishing threats.
- A new, unified approach to high availability and disaster recovery helps your business achieve increased levels of reliability.

1.3 Microsoft Exchange 2010 High Availability

For the 2010 version of Exchange, Microsoft provides a solution to most of the High Availability and load sharing requirements. The only missing piece is for that of the Client Access Server(s) (CAS). Microsoft suggests that an array of CAS can be built using a third party Load Balancer. It is this role therefore, that this document will address.
2.0 Exchange 2010 architecture
2.1 Single Client Access Server

A typical Microsoft Exchange 2010 deployment would contain a number of server roles (e.g. Hub Transport Servers, Edge Transport Servers etc.), however as these have built-in mechanisms for High Availability and load sharing these have been left out of the diagram (left) for the sake of clarity.

In this diagram we only see the Client Access Server and the Database Availability Group (containing the Mailbox Servers that the CAS connects to). The clients in this scenario are configured to access the CAS using a DNS name for the service. This name translates to the physical IP address of the CAS, should this device fails (for whatever reason), then the whole service becomes unavailable for the clients.

Also, should the number of clients accessing the Exchange service exceed the capabilities of the single CAS then the responsiveness of the service is (at best) likely to be degraded, and may possibly also fail.

The answer to both of these issues is to deploy more than one CAS, however, there then becomes an issue of how the individual clients are shared across the array of CAS.

2.2 Client Access Server Array

The answer to the problem of distributing clients across the array of CAS is to deploy a load balancing or traffic management product. In the diagram (right) this role is provided by the Stingray Traffic Managers. The Traffic Managers are deployed logically in front of the CAS array, and the clients are configured to access the service via a name that translates to an IP address.
managed by the Traffic Managers. A cluster of Traffic Managers are deployed to cope with any HA issues that may occur at this level of the infrastructure.

When a client makes a connection to their Exchange 2010 service, this connection now passes through the Stingray Traffic Manager cluster. One of the Traffic Managers will receive this connection and select a CAS for the client to be forwarded to. It makes this decision based on a number of configurable metrics, but essentially it will choose the CAS that will provide the client with the best possible user experience. In this way any performance issues or failure occurrences are dealt with transparently to the clients accessing the service.

2.3 Client Access server changes

Overview

There are a number of services that run underneath the covers of Exchange 2010.

A good number of these services use HTTPS or on occasion HTTP (TCP ports 443 and 80) for their transport, for example Outlook Web App, Exchange ActiveSync, Outlook Anywhere, and Exchange Web Services. Depending on the client software used in the environment, POP3 and IMAP4 may also be required (TCP ports 110 and 143 unencrypted, 995 and 993 under SSL).

Other Exchange services, such as the RPC Client Access service and the Exchange Address Book service, are RPC services. When an Outlook client connects directly to the Client Access server using these protocols, instead of using Outlook Anywhere, the endpoint TCP ports for these services are allocated by the RPC endpoint manager. Allocation occurs when the services are started.

This allocation is based on a “random” port being selected from a range. The configuration of the Stingray Traffic Manager requires that a node be added to the pool using the IP address and Port number, obviously if the TCP port is not known then this configuration cannot be added. Therefore, a static port mapping needs to be made for the RPC services. Once this is done then the RPC services will be restricted to port 135 and the two static ports configured via the registry. From a network perspective, nothing should need to be changed on the CAS, i.e. all network interface settings can remain the same (e.g. IP address, mask, gateway, DNS etc.). Following section of the document highlights the entire configuration necessary on Microsoft Exchange 2010 CAS server to be set up for load balancing behind Stingray Traffic Manager.

3.0 Prerequisites and Configuration tips for Microsoft Exchange 2010 Client Access Role on CAS Array

For most part this deployment guide covers the details on setting up the Riverbed’s Stingray Traffic Manager for load balancing Microsoft Exchange 2010 CAS servers. In order to make sure the Microsoft Exchange 2010 services are setup properly for load balancing most of the Exchange 2010 configuration steps are provided below. For detailed information on how to deploy or configure Microsoft Exchange 2010, refer appropriate Microsoft documentation.

3.1 Setting up Microsoft Exchange 2010 CAS Array

Microsoft Exchange 2010 CAS servers need to be setup as CAS array first so that they can be setup behind Traffic Manager for load balancing. Refer to documentation on Microsoft Exchange 2010 and here is the article on TechNet which explains the steps involved in setting up CAS array [http://blogs.technet.com/b/ucedsg/archive/2009/12/06/how-to-setup-an-exchange-2010-cas-array-to-load-balance-mapi.aspx](http://blogs.technet.com/b/ucedsg/archive/2009/12/06/how-to-setup-an-exchange-2010-cas-array-to-load-balance-mapi.aspx).

3.2 RPC Client Access (MAPI)

Configuring Static Port Mapping for RPC-Based Services

The following information is taken directly from Microsoft’s TechNet website. Located here:

By default the RPC Client Access service on an Exchange 2010 Client Access server uses the TCP End Point Mapper port (TCP/135) and the dynamic RPC port range (6005-59530) for outgoing connections, every time an Outlook client establishes a connection to Exchange. There are two static port mappings needed, the configuration of which is described below. To set a static port for the RPC Client Access service on an Exchange 2010 Client Access server, you need to open the registry on the respective server and navigate to:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\MSExchangeRPC

Here, you need to create a new key named ParametersSystem, and under this key create a REG_DWORD named TCP/IP Port. The Value for the DWORD should be the port number you want to use.

Configuring static ports for the RPC Client Access service

Microsoft recommends you set this to a unique value between 59531 and 60554 and use the same value on all CAS in any one AD site.

When you've configured the port, it's required to restart the Microsoft Exchange RPC Client Access service in order for the changes to be applied.

3.3 Exchange 2010 Address Book Service

By default the Exchange Address Book service on an Exchange 2010 Client Access server uses the TCP End Point Mapper (TCP/135) and the dynamic RPC port range (6005-59530) for outgoing connections, every time an Outlook client establishes a connection to Exchange.

Exchange 2010 RTM

In Exchange 2010 RTM a static port for the Exchange Address Book service is set using the following steps:

- Open the microsoft.exchange.addressbook.service.exe.config configuration file located in C:\Program Files\MicrosoftExchange Server\V14\Bin using Notepad.
- Change the value for the key RpcTcpPort to the port you want to use as the static port for this service. Bear in mind you cannot use the same port as you configured for the RPC Client Access service.
Configuring static port for the Exchange Address Book Service in Exchange 2010 RTM

**Note**

Microsoft recommends you set this to a unique value between 59531 and 60554 and use the same value on all Exchange 2010 Client Access servers in any one AD site.

When you've configured the port, it's required to restart the Microsoft Exchange Address Book service in order for the changes to be applied.

**Exchange 2010 SP1**

With Exchange 2010 SP1, you no longer use the "Microsoft.exchange.addressbook.service.exe.config" file to assign a static RPC port to the Exchange Address Book Service. Instead this configuration setting is controlled using the registry. To set a static RPC port for the Exchange Address Book Service, create a new REG_SZ registry key named "RpcTcpPort" under:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeAB\Parameters
```
Configuring static port for the Exchange Address Book Service in Exchange 2010 SP1

**Important**

When upgrading from Exchange 2010 RTM to SP1, you need to set this key manually after the upgrade.

**Note**

Microsoft recommends you set this to a unique value between 59531 and 60554 and use the same value on all Exchange 2010 Client Access servers in any one AD site.

When you've configured the port, it's required to restart the Microsoft Exchange Address Book service in order for the changes to be applied.

### 3.4 Exchange 2010 Public Folder connections

By default public folder connections uses the TCP End Point Mapper (TCP/135) and the dynamic RPC port range (49152-65535) for outgoing connections, every time an Outlook client establish a connection to Exchange.

**NOTE:** Exchange 2010 Public Folder Connections from the Outlook Client is directly occurs directly with Mailbox server and below configuration is not relevant for Stingray Traffic Manager configuration. Follow the below section only if you want to assign static port for public folder access from clients.

To set a static port for public folder connections, follow the same steps as those required for configuring static ports for the RPC CA service. Just bear in mind you need to perform them on the Exchange 2010 servers that stores public folder databases. This is because public folder connections from an Outlook client occur against the RPC Client Access service on the Mailbox server role.
Configuring a static port for Public Folder connections

When the port has been set for public folder connections, it's required to restart the Microsoft Exchange RPC Client Access service on the Mailbox server in order for the changes to be applied.

**Important**

Unlike in previous versions of Exchange Server, you configure static RPC ports for an Exchange 2010 Mailbox server under the `MSExchangeRPC` key and not under `MSExchangeSA\Parameters` since all MAPI connections to an Exchange 2010 Mailbox server are handled by the RPC Client Access service. For information on how to configure static RPC ports in Exchange 2007 and earlier see Microsoft KB article: Exchange Server static port mappings.

3.5 Changing the External URLs of Exchange HTTP services for respective Virtual Directories on CAS IIS Server

Below are list of EMS (Exchange Management Shell) cmdlets that can be used to setup the external URLs for all the Exchange applications.

**Outlook Web App (OWA)**

Set-OwaVirtualDirectory -Identity "CAS_Server\OWA (Default Web Site)" -ExternalURL https://mail.domain.com/OWA

**Exchange Control Panel (ECP)**

Set-EcpVirtualDirectory -Identity "CAS_Server\ECP (Default Web Site)" -ExternalURL https://mail.domain.com/ECP - FormsAuthentication $True -BasicAuthentication $True

**Exchange ActiveSync (EAS)**

Set-ActivesyncVirtualDirectory -Identity "CAS_Server\Microsoft-Server-ActiveSync (Default Web Site)" -ExternalURL https://mail.domain.com/Microsoft-Server-Activesync -BasicAuthentication $True

**Offline Address Book (OAB)**

Set-OABVirtualDirectory -Identity "CAS_Server\oab (Default Web Site)" -ExternalUrl https://mail.domain.com/oab;

**Exchange Web Services (EWS)**
Set-WebServicesVirtualDirectory -Identity "CAS_Server\EWS (Default Web Site)" -ExternalUrl https://mail.domain.com/ews/exchange.asmx

Unified Messaging (UM)

Set-UMVirtualDirectory -Identity "CAS_Server\unifiedmessaging (Default Web Site)" -InternalUrl https://mail.domain.com/unifiedmessaging/service.asmx
4.0 Stingray Traffic Manager Configuration: Separate Virtual Server For Each Microsoft Exchange 2010 CAS Client Access HTTP Service

4.1 Overview

The Traffic Manager configuration is very straightforward, there simply needs to be provision made for all the services passing through it. This means at least five Virtual Servers and Pools, with a further two if POP3 and IMAP4 need to be supported. Most of these services need to have Session Persistence configured, and specific Health Monitors setup for them. Apart from these minimal changes however, not much else needs to be changed from the default settings received Therefore, the actual Stingray Traffic Manager setup takes just a few minutes.

4.2 Configuring Stingray Traffic Manager for Outlook Web App (OWA)

This section walks through the steps required to configure Outlook Web App on dedicated/separate Traffic IP group and later portion of document will go through configuring all the Client Access HTTP services to be hosted on single Traffic IP group and single Virtual Server using Traffic Script.

1. Traffic IP Groups and Clustering
   Outlook Web app service will resolve the FQDN to Traffic IP Group address and this IP address will be active a Traffic Manager cluster. As we are talking about making Exchange 2010 highly available and scalable, it is likely that the Traffic Managers will also be deployed in a cluster (two or more Traffic Managers all active for the traffic passing through the cluster). Traffic IP Group is a configuration object that can contain a number of externally facing IP addresses, which can be raised by any member of the cluster. This spreads the traffic load across the cluster and also provides for failures within the cluster.

   Keeping the configuration simple, the minimum requirement is for one IP in the Group. In this way the Traffic Managers will act in an active-passive manner, with one of the machines in the cluster keeping the IP address raised, whilst any others stand by in readiness. Create a new Traffic IP group by accessing the WebGUI and it is located under Services → Traffic IP Groups as shown below

   ![](create_traffic_ip_group.png)
2. **Pools**
   This is an important section of the deployment for Traffic Manager as the configuration steps vary depending on whether a single pool or multiple pools for each Microsoft Exchange 2010 CAS Client Access HTTP service is chosen. Advantage of configuring dedicated pool for each Microsoft Exchange 2010 CAS Client Access HTTP service is the ability to configure different health monitors to each Client Access HTTP service. This documentation highlights the use of dedicated pool for each Client Access HTTP service. So below is the step for creating dedicated pool for Outlook Web App pool. This is done by accessing the **Services → Pools** section on WebGUI. Note that the port for nodes is set to 80 (HTTP) as the Traffic Manager is offloading SSL from servers. Also note that Full HTTP monitor is selected and later section will highlight the detailed configuration of Health Monitor. Nodes section is filled with the hostname of the CAS servers as Traffic Manager has been configured with DNS server which can resolve the hostname. If DNS server is not configured then manually you can enter the hostname to IP mappings by accessing the **System → Networking → DNS** section of Traffic Manager's WebGUI.

![Create a new Pool](image)

Though the default settings will work with the Exchange deployment there are some changes that should be made to these services. These are all at the Pool level of the configuration.

Navigate to the Pools of the new services and change the load balancing algorithms from “Round Robin” to a more useful algorithm, for example “Perceptive”. Round Robin is a nice simple and predictable algorithm for testing or very simple deployments. However it can create some issues in a production environment due to the inherent lack of intelligence of the algorithm. The Perceptive algorithm achieves a very even distribution of traffic (it uses a combination of information including current connections and responsiveness), it also has the benefit of “slow starting” recovering servers. This means that when a server fails, when it returns to the load balancing algorithm it is slowly brought up to full load. This avoids overwhelming the CAS with too much traffic.
3. **Virtual Servers**

   In order to configure the Virtual Server for OWA access the Service → Virtual Servers section of WebGUI and fill out the relevant sections under Create a new Virtual Server section.

   ![Create a new Virtual Server](image)

   Note that in the above screen the port is configured as 443 whereas protocol is HTTP. This is due to the fact that Traffic Manager will be offloading SSL from the Microsoft Exchange 2010 CAS servers. In order to configure SSL Offloading on the CAS servers for all HTTP based applications (OWA, ECP, EWS, OAB) please follow the guidelines highlighted at [http://social.technet.microsoft.com/wiki/contents/articles/how-to-configure-ssl-offloading-in-exchange-2010.aspx](http://social.technet.microsoft.com/wiki/contents/articles/how-to-configure-ssl-offloading-in-exchange-2010.aspx)

4. **Associating the Traffic IP group to configured OWA Virtual Server**

   Next step is to associate the Traffic IP group to OWA virtual server. This can be done by accessing the Services → Virtual Servers and selecting the OWA virtual server’s Basic Settings. In this section select the Listening on: section and select Traffic IP Groups radio button. Select the appropriate Traffic IP group and click update button.
SSL Key and Certificate upload for OWA

Since the Traffic Manager is offloading SSL from CAS servers, SSL Keys and Certs need to be imported to Traffic Manager. Even though Traffic Manager does have the ability to create a self-signed Certificate it is recommended to use the certificate signed by Certificate Authority. As Traffic Manager will be offloading SSL for many of Client Access HTTP services, Microsoft recommends leveraging Subject Alternate Name certificate extensions. Some examples of using SAN certificates with Exchange 2010 are shown in this TechNet Article. When you request a SAN certificate from a certification authority, you must define all desired FQDNs in the Subject Alternative Name field; clients will ignore the Common Name in the certificate Subject. Although the Traffic Manager GUI cannot create SAN certificates, and will not display the Subject Alternative Name values of imported certificates, use of SAN certificates is otherwise supported. In order to import the certificate access the Catalogs ➔ SSL ➔ SSL Certificates catalog and click on import button to upload the certificate and key file.
6. **Enable SSL decrypt and associate SSL Certificate to Virtual Server**
   Next step once the SSL Key and Certs are uploaded, associate it to the virtual server and enable SSL decrypt on the virtual server. This is done by accessing the appropriate virtual server under **Services → Virtual Servers**. Click SSL Decryption section under the virtual server and select the appropriate certificate that was uploaded. Check the `ssl_decrypt` setting to **Yes** and click update button at the bottom of the page. Now the Virtual Server is enabled for SSL Decryption.

7. **Create a OWA Persistence Class**
   Microsoft recommends use of cookie persistence for OWA application and refer to the affinity options at [http://technet.microsoft.com/en-us/library/ff625247.aspx](http://technet.microsoft.com/en-us/library/ff625247.aspx) for Microsoft’s recommendation for setting up persistence for different Client Access services when using ADCs like Traffic Manager. Setup persistence class of type Transparent Session affinity persistence by accessing **Catalogs → Persistence**, create a new class with name and press the create class button and then select Transparent session affinity and click update button to apply changes.
8. **Create Health Monitor for OWA**
   As it is recommended to have separate pool for each Client Access HTTP service, create a specific Health Monitor for OWA by accessing **Catalogs → Monitors**
This completes the configuration for separate virtual servers for the OWA Client Access service.
9. **Associate Session Persistence and Health Monitor to OWA Pool**

Access the Services → Pools, and select the OWA pool that was configured earlier in step 2. Click on Session Persistence section and select the persistence class created earlier for OWA.
Next select the Monitors section under the OWA pool and select the appropriate monitor from the drop down menu under Add monitor: section.

4.3 Configuring Stingray Traffic Manager to Redirect All HTTP requests to SSL

As Traffic Manager will only be handling SSL traffic, clients trying to access the applications on port HTTP should be redirected to connect back on SSL. Following steps walks through configuration of virtual server with a traffic script which will redirect all the clients trying to connect on port http.

1. Create a Virtual Server with Traffic Pool set to Discard
   Access the WebGUI and create new virtual server by navigating to Services → Virtual Servers
2. **Create a Traffic Script to redirect to proper SSL URL**

Access the WebGUI and create a Traffic Script by navigating to Catalogs → Rules and create the following Traffic Script. The traffic script is available in appendix section for copying the code and modifying.
3. **Associating the redirect TrafficScript to the Virtual Server**

Traffic Script created in step 2 needs to be associated with the virtual server created in step 1. This can be done by navigating to **Services → Virtual Servers**, selecting the appropriate virtual server created in step 1 and selecting the traffic script created in step 2 from the drop down menu for **Add Rule** under Request Rules section and click Add Rule button.

This completes creating a redirect Virtual Server.
4.4 Configuring Stingray Traffic Manager for Outlook Anywhere

Outlook Anywhere for Exchange 2010 allows you to use Microsoft Outlook clients to connect to your Exchange server over the Internet, using HTTPS to encapsulate RPC (MAPI) traffic.

**Important**

To enable and require SSL for all communications between the Client Access server and the Outlook clients, trusted certificate signed by Certificate Authority should be obtained and published at the default Web site level. It is recommended that the certificate be purchased from a third-party certification authority whose certificates are trusted by a wide variety of Web browsers. By default, applications and Web browsers do not trust root certification authority when there is internal/non-trusted certification authority, such as a Stingray Traffic Manager self-signed certificate. When a user tries to connect to Microsoft Outlook by using Outlook Anywhere, and the user's computer does not trust the certificate and root Certificate Authority, the connection fails. For more information on this topic, see the following Microsoft TechNet article: [http://technet.microsoft.com/en-us/library/aa997703.aspx](http://technet.microsoft.com/en-us/library/aa997703.aspx).

Following are the steps to create an Outlook Anywhere Virtual Server tied to dedicated Traffic IP group:

1. **Create Traffic IP group that is mapped to FQDN of Outlook Anywhere service**

   ![Create Traffic IP Group](image)

   - Name: DA.company.com
   - Traffic Managers: SRTM1
     - 10.32.147.125
   - IP Addresses: 192.168.22.102

   **Create Traffic IP Group**

2. **Create a Pool for Outlook Anywhere Service**

   ![Create a new Pool](image)

   - Pool Name: Exchange 2010 Outlook Anywhere Pool
   - Nodes: CAS-1:80 CAS-2:80
   - Use Auto-Scaling for the nodes in this pool
   - Monitor: No Monitor

   **Create Pool**

3. **Create Virtual Server for Outlook Anywhere Service**

   Note that the Port is set to 443 as the Virtual server will be offloading the SSL from the CAS servers. We will use the same certificate that we uploaded for the OWA service to take advantage of the Subject Alternative Names. Make sure that the FQDN DNS name is part of the SAN Field.
4. **Associate Traffic IP Group to Outlook Anywhere Virtual Server**

In the Virtual Server configuration associate the Traffic IP Group created in step 1 to OA virtual server created in step 3.

5. **SSL Key and Certificate upload for Outlook Anywhere**

Since the Traffic Manager is offloading SSL from CAS servers, SSL Keys and Certs need to be imported to Traffic Manager. Even though Traffic Manager does have the ability to create a self-signed Certificate it is recommended to use the certificate signed by Certificate Authority. As Traffic Manager will be offloading SSL for many of Client Access HTTP services, Microsoft recommends leveraging Subject Alternate Name certificate extensions. Some examples of using SAN certificates with Exchange 2010 are shown in this [TechNet Article](https://technet.microsoft.com). When you request a SAN certificate from a certification authority, you must define all desired FQDNs in the Subject Alternative Name field; clients will ignore the Common Name in the certificate Subject. Although the Traffic Manager GUI cannot create SAN certificates, and will not display the Subject Alternative Name values of imported certificates, use of SAN certificates is otherwise supported. In order to import the certificate access the **Catalogs** → **SSL** → **SSL Certificates catalog** and click on import button to
upload the certificate and key file.

6. **Enable SSL decrypt and associate SSL Certificate to Virtual Server**
   Next step once the SSL Key and Certs are uploaded, associate it to the virtual server and enable SSL decrypt on the virtual server. This is done by accessing the appropriate virtual server under Services \> Virtual Servers. Click SSL Decryption section under the virtual server and select the appropriate certificate that was uploaded. Check the ssl_decrypt setting to Yes and click update button at the bottom of the page. Now the Virtual Server is enabled for SSL Decryption.

7. **Configuring Persistence Class and Create a Traffic Scrip for Persistence for Outlook Anywhere service**
   Outlook Anywhere requires persistence as clients split the RPC connections into two (RPC_IN_DATA and RPC_OUT_DATA). If the CAS servers are behind a load balancer, then load balancer needs to make sure that both connections are sent to same CAS server. Traffic Manager’s Universal Persistence type is used to create persistence records based on the type of the client accessing the service. The reason for tracking different type of Outlook client is that some of the older versions of Outlook clients (prior to Outlook 2010) doesn’t support OutlookSession cookie. So for these older version of Outlook clients the value of HTTP header Authorization is used to create presistence records and for Outlook 2010 persistence records are created using OutlookSession cookie. The text of the traffic script is in the appendix section for copy and modification.
Persistence configuration steps highlighted in this document is for BASIC AUTH authentication method and not NTLM

**Create new Session Persistence class**

<table>
<thead>
<tr>
<th>Name:</th>
<th>Exchange 2010 Outlook</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Create Class</td>
</tr>
</tbody>
</table>

**Class: Exchange 2010 Outlook Anywhere Persistence**

- **Pools using this:** none
- **Last Modified:** 16 Dec 2011 13:28

### Basic Settings

Each Session Persistence class controls two main issues: How to identify requests from the same session, and what action to take when a session expires.

**Name:** Exchange 2010 Outlook Anywhere

- The type of session persistence to use:
  - **IP-based persistence**
    - Send all requests from the same source address to the same node.
  - **Universal session persistence**
    - Use session persistence data supplied by a TrafficScript rule.
  - **Named Node session persistence**
    - Use a node specified by a TrafficScript rule.
  - **Transparent session affinity**
    - Insert cookies into the response to track sessions.
  - **Monitor application cookies**
    - Monitor a specified application cookie to identify sessions.
  - **J2EE session persistence**
    - Monitor Java's JSESSIONID cookie and URLs.
  - **ASP and ASP.NET session persistence**
    - Monitor ASP session cookies and ASP.NET session cookies and cookieless URLs.
  - **X-Zeus-Backend cookies**
    - Inspect an application cookie named "X-Zeus-Backend" which names the destination node.
  - **SSL Session ID persistence**
    - Use the SSL Session ID to identify sessions (SSL pass through only).

**Create new rule**

<table>
<thead>
<tr>
<th>Name:</th>
<th>Exchange 2010 Outlook</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Use RuleBuilder</td>
</tr>
<tr>
<td></td>
<td>Use TrafficScript Language</td>
</tr>
<tr>
<td></td>
<td>Create Rule</td>
</tr>
</tbody>
</table>
8. **Associate the Traffic Script created in Step 5 with the Outlook anywhere Virtual Server**
   Select the rule created in step5 from the drop down menu for **Add Rule** : under the Request Rules section and click add rule button.

   ![TrafficScript code](image)

   ```
   $universal_session_persistence = "Exchange 2010 Outlook Anywhere Persistence";
   $outlooksession = http.getCookie( "OutlookSession" );
   if ( $outlooksession ) {
       connection.setPersistence( $universal_session_persistence);
       connection.setPersistenceKey( $outlooksession);
   }
   #Create Persistence records for all other clients based on the value of Authorization Header
   close {
       connection.setPersistence( $universal_session_persistence);
       connection.setPersistenceKey( $auth);
   }
   ```

TrafficScript rules are evaluated in order. If a rule selects a pool, the request is balanced by that pool. **No rules have been configured for this virtual server**

**Add rule:** Exchange 2010 Outlook Anywhere Persistence
10. **Create Health Monitor for Outlook Anywhere**

As it is recommended to have separate pool for each Client Access HTTP service, create a specific Health Monitor for Outlook Anywhere by accessing **Catalogs → Monitors**

<table>
<thead>
<tr>
<th>Name: Exchange 2010 OA Monitor</th>
</tr>
</thead>
</table>

**The internal monitor implementation of this monitor.**

- **type:**
  - Ping monitor
  - TCP Connect monitor
  - HTTP monitor
  - **TCP transaction monitor**
  - External program monitor ...
  - SIP monitor
  - RTSP monitor

**A monitor can either monitor each node in the pool separately and disable an individual node if it fails. GLB location monitors must monitor a specific machine.**

- **scope:**
  - Node: Monitor each node in the pool separately
  - Pool/GLB: Monitor a specified machine ...

**Create Monitor**

Use of the TCP transaction monitor enables Stingray to send a customer URL string. Since there are no authentication headers in the request sent to the server, the expected response is to get a page which states “You do not have permission to access this page”. Also external Perl scripts can be written and associated with this monitor. At minimum you can set the monitor to TCP connect monitor if just Layer4 TCP monitor if need be.
4.5 Configuring Stingray Traffic Manager for ActiveSync

Exchange ActiveSync is a synchronization protocol based on HTTP and XML that is designed to work over a cellular, wireless Internet or other similar low-bandwidth, high-latency connections. Exchange ActiveSync can synchronize e-mail messages, contacts, calendar, and task data.

This completes the configuration of Outlook Anywhere Service on Traffic Manager.
1. **Create Traffic IP Group for ActiveSync Service**

   **Create a new Traffic IP Group**
   
<table>
<thead>
<tr>
<th>Name:</th>
<th>ActiveSync.company.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic Managers:</td>
<td>Traffic Manager</td>
</tr>
<tr>
<td>SRTM1</td>
<td>10.32.147.125</td>
</tr>
<tr>
<td>IP Addresses:</td>
<td>192.168.22.103</td>
</tr>
</tbody>
</table>

   **Create Traffic IP Group**

2. **Create Pool for ActiveSync**

   **Create a new Pool**
   
<table>
<thead>
<tr>
<th>Pool Name:</th>
<th>Exchange 2010 ActiveSync Pool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nodes:</td>
<td>CAS-1:80 CAS-2:80</td>
</tr>
<tr>
<td>Use Auto-Scaling for the nodes in this pool</td>
<td>☑</td>
</tr>
<tr>
<td>Monitor:</td>
<td>Full HTTP</td>
</tr>
</tbody>
</table>

   **Create Pool**

3. **Create Virtual Server for ActiveSync**

   **Create a new Virtual Server**
   
<table>
<thead>
<tr>
<th>Virtual Server Name:</th>
<th>ActiveSync.company.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol:</td>
<td>HTTP</td>
</tr>
<tr>
<td>Port:</td>
<td>443</td>
</tr>
<tr>
<td>Default Traffic Pool:</td>
<td>Exchange 2010 ActiveSync Pool</td>
</tr>
</tbody>
</table>

   **Create Virtual Server**

4. **Associate the Traffic IP group created in Step 1 with Virtual Server created in Step 3**
5. **SSL Key and Certificate upload for ActiveSync**

Since the Traffic Manager is offloading SSL from CAS servers, SSL Keys and Certs need to be imported to Traffic Manager. Even though Traffic Manager does have the ability to create a self-signed Certificate it is recommended to use the certificate signed by Certificate Authority. As Traffic Manager will be offloading SSL for many of Client Access HTTP services, Microsoft recommends leveraging Subject Alternate Name certificate extensions. Some examples of using SAN certificates with Exchange 2010 are shown in this TechNet Article. When you request a SAN certificate from a certification authority, you must define all desired FQDNs in the Subject Alternative Name field; clients will ignore the Common Name in the certificate Subject. Although the Traffic Manager GUI cannot create SAN certificates, and will not display the Subject Alternative Name values of imported certificates, use of SAN certificates is otherwise supported. In order to import the certificate access the **Catalogs → SSL → SSL Certificates catalog** and click on import button to upload the certificate and key file.
SSL Certificates Catalog

6. **Enable SSL decrypt and associate SSL Certificate to Virtual Server**

Next step once the SSL Key and Certs are uploaded, associate it to the virtual server and enable SSL decrypt on the virtual server. This is done by accessing the appropriate virtual server under **Services → Virtual Servers**. Click SSL Decryption section under the virtual server and select the appropriate certificate that was uploaded. Check the `ssl_decrypt` setting to **Yes** and click update button at the bottom of the page. Now the Virtual Server is enabled for SSL Decryption.

7. **Create Persistence class of type universal persistence and traffic script for persistence creation**

Steps of creating the persistence class and traffic script are shown below. Appendix will contain the code for this traffic script for copying and modification.
Create new Session Persistence class

Name: Exchange 2010 ActiveSync Persistence

Class: Exchange 2010 ActiveSync Persistence

Pools using this: none

Last Modified: 16 Dec 2011 13:33

Basic Settings

Each Session Persistence class controls two main issues: How to identify requests from the same session, and what action to take once identification is successful.

Name: Exchange 2010 ActiveSync Persistence

The type of session persistence to use.

- IP based persistence
  Send all requests from the same source address to the same node.

- Universal session persistence
  Use session persistence data supplied by a TrafficScript rule.

- Named Node session persistence
  Use a node specified by a TrafficScript rule.

- Transparent session affinity
  Insert cookies into the response to track sessions.

- Monitor application cookies...
  Monitor a specified application cookie to identify sessions.

- J2EE session persistence
  Monitor Java's JSESSIONID cookie and URLs

- ASP and ASP.NET session persistence
  Monitor ASP session cookies and ASP.NET session cookies and cookiless URLs.

- X-Zeus-Backend cookies
  Inspect an application cookie named 'X-Zeus-Backend' which names the destination node.

- SSL Session ID persistence
  Use the SSL Session ID to identify sessions (SSL pass-through only).

Create new rule

Name: Exchange 2010 ActiveSync Persistence

- Use RuleBuilder

- Use TrafficScript Language

Create Rule
Important

Persistence configuration steps highlighted above in this document is not applicable when SSL Client Certificate Authentication is enabled.

8. Associate the rule created in step 5 to ActiveSync Virtual Server
Creating Health Monitor for ActiveSync

As it is recommended to have a separate pool for each Client Access HTTP service, create a specific Health Monitor for ActiveSync by accessing Catalogs → Monitors.

**Virtual Server: ActiveSync.company.com (HTTP, port 443)**

TrafficScript rules are evaluated in order. If a rule selects a pool, the request is balanced by that pool.

**Request Rules**

Request rules are evaluated before the request is sent to the pool.

*No rules have been configured for this virtual server*

**Add rule:** Exchange 2010 ActiveSync Persistence

9. **Creating Health Monitor for ActiveSync**

As it is recommended to have a separate pool for each Client Access HTTP service, create a specific Health Monitor for ActiveSync by accessing Catalogs → Monitors.

**Create new monitor**

**Name:** Exchange 2010 ActiveSync

The internal monitor implementation of this monitor.

**type:**

- Ping monitor
- TCP Connect monitor
- HTTP monitor
- TCP transaction monitor
- External program monitor...
- SIP monitor
- RTSP monitor

A monitor can either monitor each node in the pool separately and disable an individual node if it fails, machine fails. GLB location monitors must monitor a specific machine.

**scope:**

- Node: Monitor each node in the pool separately
- Pool/GLB: Monitor a specified machine...
Fill in the FQDN of ActiveSync service and since the request is not sending any authentication credentials the expected response is "Access is denied" page which is what the response_regex is catching to mark the health of the server. Advanced external monitors can be written in any language of choice and be associated with the pool.

This completes the creation of ActiveSync Virtual Server on Traffic Manager.
4.6 Configuring Stingray Traffic Manager for Auto Discover

The Autodiscover service provides automatic configuration information to recent versions of Outlook and some mobile clients. Autodiscover service doesn’t need any kind of persistence.

**Important**


1. **Create Traffic IP Group for Auto Discover Service**

   ![Create Traffic IP Group](image)

   - **Name:** autodiscover.company.com
   - **Traffic Managers:**
     - **Add:** SRTM1
     - **IP Addresses:** 192.168.22.104

   [Create Traffic IP Group]

2. **Create Pool for Active Discover**

   ![Create Pool](image)

   - **Pool Name:** Exchange 2010 Auto Discover Pool
   - **Nodes:** CAS-1:80 CAS-2:80
   - **Monitor:** Client First

   [Create Pool]

3. **Create Virtual Server for Auto Discover**

   Virtual server will be offloading SSL and will use the same certificate that was used for OWA service which has SAN DNS name resolving to autodiscover traffic ip group. As per this deployment guide it will be the IP address configured in step 1.
4. **SSL Key and Certificate upload for Auto Discover**

   Since the Traffic Manager is offloading SSL from CAS servers, SSL Keys and Certs need to be imported to Traffic Manager. Even though Traffic Manager does have the ability to create a self-signed Certificate it is recommended to use the certificate signed by Certificate Authority. As Traffic Manager will be offloading SSL for many of Client Access HTTP services, Microsoft recommends leveraging Subject Alternate Name certificate extensions. Some examples of using SAN certificates with Exchange 2010 are shown in this TechNet Article. When you request a SAN certificate from a certification authority, you must define all desired FQDNs in the Subject Alternative Name field; clients will ignore the Common Name in the certificate Subject. Although the Traffic Manager GUI cannot create SAN certificates, and will not display the Subject Alternative Name values of imported certificates, use of SAN certificates is otherwise supported. In order to import the certificate access the **Catalogs ➔ SSL ➔ SSL Certificates catalog** and click on import button to upload the certificate and key file.
5. **Enable SSL decrypt and associate SSL Certificate to Virtual Server**

Next step once the SSL Key and Certs are uploaded, associate it to the virtual server and enable SSL decrypt on the virtual server. This is done by accessing the appropriate virtual server under Services ➔ Virtual Servers. Click SSL Decryption section under the virtual server and select the appropriate certificate that was uploaded. Check the ssl_decrypt setting to Yes and click update button at the bottom of the page. Now the Virtual Server is enabled for SSL Decryption.

6. **Create Health Monitor for Auto Discover and Associate it to the Auto Discover Pool**

As it is recommended to have separate pool for each Client Access HTTP service, create a specific Health Monitor for Auto Discover by accessing Catalogs ➔ Monitors. You can use the default Full HTTP monitor on a default page or can write more complex monitor using Perl or any other customer script. This script can then be associated to the pool via external program monitor option.

This completes the Auto Discover configuration.

### 4.7 Configuring Stingray Traffic Manager for MAPI RPC Client Access

Outlook Clients which use native MAPI, access the service via CAS servers in Exchange 2010 which is an architectural change from earlier versions of Exchange. Stingray Traffic Manager can load balance the native MAPI access to CAS Servers. MAPI RPC client access connects over large range of dynamically negotiated ports. Since Traffic Manager doesn’t natively support...
virtual servers to listen on range of ports this section of MAPI configuration is based on static mapping of Mailbox and Address Book ports on CAS Servers as mentioned in the perquisites section for configuration of Microsoft Exchange 2010 CAS servers. There will be three virtual servers configured

a. Virtual Server for MAPI End Point Mapper service on TCP port 135
b. Virtual Server for Mailbox Access service on TCP port 55001 (randomly chosen based on Microsoft’s recommendation)
c. Virtual Server for Address Book service on TCP port 55003 (randomly chosen based on Microsoft’s recommendation)

All the Virtual Servers will be associated with same Traffic IP group and same persistence class of type IP persistence

1. **Create Traffic IP Group for RPC Client Access**

   ![Traffic IP Group](image)

   a. Create Traffic IP Group for RPC Client Access

   ![Traffic IP Group](image)

   b. Create Pool for MAPI End Point Mapper Service with Monitor type connect

   ![Create a new Pool](image)

   3. **Set Pool connection management settings to handle Node failure conditions**

   ![Pool Settings](image)

   4. **Create Virtual Server for MAPI End Point Mapper Service with Generic Client first protocol and associate the with Traffic IP group created in step 1**
5. Set Timeout settings
As per Microsoft KB article [http://support.microsoft.com/kb/2535656](http://support.microsoft.com/kb/2535656), set the connection timeout settings under connection management to 7200 seconds.

6. Create persistence class of type IP persistence
Create new Session Persistence class

Name: **RFC Persistence**

Create Class

**Basic Settings**

Each Session Persistence class controls two main issues: How to identify requests from the same session and what type of session persistence to use.

**Name:** RFC Persistence

**Type:**
- **IP-based persistence**
  - Send all requests from the same source address to the same node.
- **Universal session persistence**
  - Use session persistence data supplied by a TrafficScript rule.
- **Named Node session persistence**
  - Use a node specified by a TrafficScript rule.
- **Transparent session affinity**
  - Insert cookies into the response to track sessions.
- **Monitor application cookies**
  - Monitor a specified application cookie to identify sessions.

7. Associate the persistence class created in step 4 to MAPI End Point Mapper Service Pool

**Pool:** CAS-RPC-135 (Generic client first, 2 nodes)

Session Persistence ensures that all requests from a client will always get sent to the same node.

**Session Persistence Catalog**

Choose Session Persistence Class

The default Session Persistence class this pool uses, if any:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>IP-based persistence</td>
</tr>
<tr>
<td>Exchange 2010 ActiveSync Persistence</td>
<td>IP-based persistence</td>
</tr>
<tr>
<td>Exchange 2010 Outlook Anywhere Persistence</td>
<td>IP-based persistence</td>
</tr>
<tr>
<td>Exchange Cookie Insert Persistence</td>
<td>Transparent session affinity</td>
</tr>
<tr>
<td>Exchange Universal Session Persistence</td>
<td>Universal session persistence</td>
</tr>
<tr>
<td>RPC Persistence</td>
<td>IP-based persistence</td>
</tr>
</tbody>
</table>

Update
8. **Create Pool for MAPI Mailbox Service with Monitor type connect**

![Create a new Pool](image)

- **Pool Name:** CAS-RPC-55001
- **Nodes:** CAS-1:55001 CAS-2:55001
- **Monitor:** Client First

9. **Set Pool connection management settings to handle Node failure conditions**

The number of times the software will attempt to connect to the same back-end node before marking it as failed. This is only used when passive_monitoring is enabled:

- **node_connection_attempts:** 3

The amount of time, in seconds, that a traffic manager will wait before re-trying a node that has been marked as failed by passive monitoring:

- **node_fail_time:** 20 seconds

- **node_conn_close:** Yes

10. **Create Virtual Server for MAPI Mailbox Service with Generic Client first protocol. And Associate the traffic ip group created step 1**

![Create a new Virtual Server](image)

- **Virtual Server Name:** MAPI RPC _Mailbox
- **Protocol:** Generic client first
- **Port:** 55001
- **Default Traffic Pool:** CAS-RPC-55001
11. **Set Timeout settings**
   As per Microsoft KB article [http://support.microsoft.com/kb/2535656](http://support.microsoft.com/kb/2535656), set the connection timeout settings under connection management to 7200 seconds.

12. **Associate the persistence class created in step 5 to MAPI End Point Mapper Service Pool**
13. Create Pool for MAPI Address Book Service with Monitor type connect

14. Set Pool connection management settings to handle Node failure conditions

15. Set Pool connection management settings to handle Node failure conditions
   And Associate the traffic IP group created step 1

17. Set Timeout settings
    As per Microsoft KB article [http://support.microsoft.com/kb/2535656](http://support.microsoft.com/kb/2535656), set the connection timeout settings under
connection management to 7200 seconds.

18. Associate the persistence class created in step 5 to MAPI Address Book Service Pool

This completes the configuration of MAPI RPC client access service.

4.8 Configuring Stingray Traffic Manager for POP3

POP3 Service on Exchange CAS servers enables mail clients which support POP3 protocol to access Exchange CAS servers running POP3 service. There are variety of clients including Outlook, Outlook Express, Eudora and other 3rd party clients. Instructions in this guide details the configuration steps on Traffic Manager to service POP3S, and process all the secure traffic and forward unencrypted traffic to exchange servers running POP3 service.

For more information about how to manage POP3 in Exchange 2010, see Understanding POP3 and IMAP4 on Microsoft TechNet at http://technet.microsoft.com/en-us/library/bb124107%28EXCHG.140%29.aspx

1. Create Traffic IP Group for POP3 Service
2. Create Pool for POP3 and associate Monitor type POP

3. Create Virtual Server for POP3 to listen on port 995 with protocol set to POP3 and associate to Traffic IP group created in step 1
4. Enable SSL Decryption using the same certificate/key pair used for OWA as it supports SAN DNS Name for POP3 service. If there is a dedicated certificate/key pair for POP3 service, then follow the normal procedure of importing the Key and Certificate and use the same procedure as shown below to enable SSL decryption.

This completes the configuration of POP3.
4.9 Configuring Stingray Traffic Manager for IMAP4

Same as POP3 Service, IMAP4 service enable mail clients which support IMAP4 protocol to access Exchange CAS servers running IMAP4 service. There are variety of clients including Outlook, Outlook Express, Eudora and other 3rd party clients. These instructions in this guide details the configuration steps on Traffic Manager to service IMAP4S, and process all the secure traffic and forward unencrypted traffic to exchange servers running IMAP4 service.

For more information about how to manage POP3 in Exchange 2010, see Understanding POP3 and IMAP4 on Microsoft TechNet at http://technet.microsoft.com/en-us/library/bb124107%28EXCHG.140%29.aspx

1. Create Traffic IP Group for IMAP4 Service or you can use the same Traffic IP that was created for POP3 service if the FQDN for service resolves to same IP as POP3 Traffic Group IP address. This steps assumes that the POP3 and IMAP4 FQDN names resolves to same Traffic IP address.

2. Create Pool for IMAP4 and associate Monitor type Connect

3. Create Virtual Server for IMAP4 to listen on port 993 with protocol set to IMAP4 and associate to Traffic IP group created in step 1.
4. Enable SSL Decryption using the same certificate/key pair used for OWA as it support SAN DNS Name for IMAP4 service. If there is a dedicated cert/key pair for IMAP4 service then follow the normal procedure of importing the Key and Certificate and use the same procedure as shown below to enable ssl decryption.


If you chose a single virtual server for all HTTP-based services, then following are the detailed configuration steps on Traffic Manager.

1. Create Traffic IP Group that is mapped to FQDN of all the services.
   In this document all the services will resolve to webmail.company.com traffic ip group which is 192.168.22.101. Now all
services, For example: activesync.company.com, oa.company.com, autodiscover.company.com, Webmail.company.com all resolve to 192.168.22.101 Traffic IP address.

2. **Create the following Pools as detailed in previous sections of this document and do not configure any persistence for the Pools**
   a. Exchange 2010 Outlook Web Access Pool
   b. Exchange 2010 Outlook Anywhere Pool
   c. Exchange 2010 ActiveSync Pool
   d. Exchange 2010 Auto Discover Pool

3. **Create Virtual Server**
   Following step creates a Virtual server named webmail.company.com, associates the Exchange 2010 Outlook Web Access Pool to the virtual server and associates Traffic IP group created in step 1. This would be the default pool for this virtual server and traffic script will direct the traffic to appropriate pool based on the URLs.

4. **Enable SSL Decryption on the Virtual Server and use the certificate which SAN DNS Names for all the Exchange Web applications.**
5. **Create Two Persistence classes**
   a. Persistence class of type Universal Session Persistence and in this document it is configured with name “Exchange Universal Session Persistence”. This persistence class will hold all the records other than Outlook Web App persistence records.

   **Class: Exchange Universal Session Persistence**

   Pools using this: *none*

   Last Modified: 14 Dec 2011 11:54

   **Basic Settings**

   Each Session Persistence class controls two main issues: How to identify requests from the same...

   Name: Exchange Universal Session Persistence

   The type of session persistence to use.

   - **IP-based persistence**: Send all requests from the same source address to the same node.
   - **Universal session persistence**: Use session persistence data supplied by a TrafficScript rule.

   b. Persistence class of type “Transparent Session Affinity” and in this document it is configured with name “Exchange Cookie Insert Persistence”.

   ```
   ```
6. **Create a Traffic Script that forwards the requests to appropriate pool and also creates appropriate persistence records.**

In this document the Traffic Script rule is named “Exchange2010 Single TrafficIP and Virtual Server for All HTTP Applications”. The code is available in the appendix section for copying and modification.
7. Associate the Traffic Script rule with the virtual server created in Step 3.
This completes the configuration of single virtual server on Traffic Manager to handle all Exchange 2010 Client Access services.

### 6.0 Configuration Summary Of All Microsoft Exchange 2010 Services on Traffic Manager

By accessing the Services ➔ Config Summary on the webGUI a complete snapshot of all the configured services is provided. This is very useful table to glance through to get a good understanding of how the services are configured.

<table>
<thead>
<tr>
<th>Virtual Servers</th>
<th>Rules</th>
<th>Pools</th>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveSync.company.com:443</td>
<td>Use default pool</td>
<td></td>
<td></td>
</tr>
<tr>
<td>imap.company.com:993</td>
<td>Use default pool</td>
<td>CAS-IMAP4</td>
<td></td>
</tr>
<tr>
<td>MAPI RPC - Address Book cas.company.com:55003</td>
<td>Use default pool</td>
<td>CAS-RPC-55003</td>
<td></td>
</tr>
<tr>
<td>MAPI RPC - Mailbox cas.company.com:55001</td>
<td>Use default pool</td>
<td>CAS-RPC-55001</td>
<td></td>
</tr>
<tr>
<td>MAPI RPC Endpoint Mapper cas.company.com:135</td>
<td>Use default pool</td>
<td>CAS-RPC-135</td>
<td></td>
</tr>
<tr>
<td>pop.company.com:993</td>
<td>Use default pool</td>
<td>CAS-POP</td>
<td></td>
</tr>
<tr>
<td>Webmail.company.com:443</td>
<td>Use default pool</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Webmail.company.com:Redirect Webmail.company.com:80</td>
<td>OWA_Redirect_SSL</td>
<td>discarded</td>
<td></td>
</tr>
</tbody>
</table>
### 7.0 Configuration Worksheet

<table>
<thead>
<tr>
<th>Virtual Server Name: Port</th>
<th>Pool Nodes</th>
<th>SSL</th>
<th>Type of Persistence</th>
<th>TrafficScript for Persistence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of Virtual Server:</td>
<td>1 Key:</td>
<td></td>
<td>Transparent Session Affinity (Cookie Insert)</td>
<td>Not Needed</td>
</tr>
<tr>
<td>Internet Protocol: HTTP</td>
<td>2 Cert:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Port: 443 (if SSL offloaded)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name of Virtual Server:</td>
<td>1 Key:</td>
<td></td>
<td>Universal Session Persistence</td>
<td>Need for creating Universal Session Persistence</td>
</tr>
<tr>
<td>Internet Protocol: HTTP</td>
<td>2 Cert:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Port: 443 (if SSL offloaded)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name of Virtual Server:</td>
<td>1 Key:</td>
<td></td>
<td>Persistence not needed</td>
<td></td>
</tr>
<tr>
<td>Internet Protocol: HTTP</td>
<td>2 Cert:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Port: 443 (if SSL offloaded)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| RPC End Point Mapper Service | IP Based Persistence |
| Name of Virtual Server: | 2                      |
| Internet Protocol: Generic Client First | 3                      |
| Port: 135 | 4                      |

| Mailbox Service | IP Based Persistence |
| Name of Virtual Server: | 2                      |
| Internet Protocol: Generic Client First | 3                      |
| Port: Statically configured Port for Mailbox on CAS Server | 4                      |

| Address Book Service | IP Based Persistence |
| Name of Virtual Server: | 2                      |
| Internet Protocol: Generic Client First | 3                      |
| Port: Statically configured Port for Addressbook on CAS Server | 4                      |

| Name of Virtual Server: | 1 Key: (If POP3S) | Persistence not needed |
| Internet Protocol: POP3 | 2 Cert: (If POP3S) |                              |
| Port: 995 (if POP3S else 110) | 3     |                              |

| Name of Virtual Server: | 1 Key: (If IMAPv4S) | Persistence not needed |
| Internet Protocol: IMAPv4 | 2 Cert: (If IMAPv4S) |                              |
| Port: 993 (if IMAPv4S else 143) | 3     |                              |

### 8.0 Extra Optional Functionality

The Stingray Traffic Manager is much more than just a simple load balancer, therefore there are a number of other functions/features that you may wish to deploy with your Exchange 2010 CAS. These have been described in this separate section as they are not necessary, but could enhance the performance or manageability of your environment. Further descriptions of all these features can be found in the STM User Manual.
• Service Level Monitoring – this feature monitors the responses of your CAS and can send alerts should these fall below an expected threshold of performance
• Global Load Balancing – this enables clients to be distributed across multiple locations, either for DR purposes or based on their geographic proximity to a datacenter.

9.0 Implementation

9.1 Physical Network Deployment
As a Reverse Proxy, the deployment options for the Stingray Traffic Manager are extremely flexible. In most instances there are no changes required to the existing network infrastructure. The STM will simply be added to the network and traffic directed to it via DNS.

There is a whole chapter in the User Manual that addresses this aspect of the deployment, it is suggested that you reference this for a complete understanding.

9.2 Domain Name Service
As has been mentioned previously, traffic that would have been sent directly to the CAS before the deployment of the Traffic Managers now needs to terminate at the STM. This is quite easy to achieve, the zone files for the domain will need to be altered. The A Records that relate the name of the service to the IP address of the CAS now need to point to the Traffic IPs of the Traffic Manager.

These changes can take some time to become effective in every location (due to caching of previous results). Testing prior to the move of the IPs can be done by using static mappings in the clients host file, or by using the IP address of the Traffic Manager only.

10.0 Monitoring
The Stingray Traffic Manager has some great tools to assist in managing and monitoring your online application traffic. These can be accessed via the web UI of the device, via the Activity tab.

Real-time graphing can be used to show the traffic passing through the STM in a very granular way, you can change the data being monitored down to an individual node, or see all the traffic for the entire deployment.

There are also a map view, and connection list to aid further visibility of the traffic.

11.0 CONCLUSION
This document briefly discusses how to configure Stingray Traffic Manager to effectively load balance traffic to a farm of Microsoft Exchange 2010 Client Access Servers. Stingray Traffic Manager is able to manage traffic in a wide variety of ways, to improve the performance, security, reliability and integrity. Please refer to the product documentation on the Riverbed Community Forums (http://community.riverbed.com) for examples of how Stingray Traffic Manager can be deployed to meet a range of service hosting problems.

APPENDIX

1. Traffic Script Code for redirecting all HTTP requests to SSL referenced as “OWA_Redirect_SSL” in section 4.3

```bash
# Redirect to OWA url if user tries default website
$hostheader = http.getHostHeader();
if (http.getPath() == "/") {
    http.redirect("https://".$hostheader."/owa");
}
```

2. Traffic Script code for Outlook Anywhere Persistence referenced as " Exchange 2010 Outlook Anywhere"
# Extract the value of Authorization Header and OutlookSession cookie

```powershell
defined as "Exchange 2010 Outlook Anywhere Persistence"

# Validating if the Cookie named OutlookSession exists and has value to track Outlook 2010 clients and Create persistence based on the Cookie value as the Key

if ( $outlooksession ) {
    connection.setPersistence( $universal_session_persistence );
    connection.setPersistenceKey( $outlooksession );
} # Create Persistence records for all other clients based on the value of Authorization Header

else {
    connection.setPersistence( $universal_session_persistence );
    connection.setPersistenceKey( $auth );
}
```


```powershell
# Collect the value of Authorization Header and OutlookSession cookie

$auth = http.getHeader( "Authorization" );

# Make sure to declare a persistence class which matches below name

$universal_session_persistence = "Exchange 2010 ActiveSync Persistence";

# Create Persistence records based on the value of Authorization Header

if ( $auth ) {
    connection.setPersistence( $universal_session_persistence );
    connection.setPersistenceKey( $auth );
}

```
# Please declare the names of the pools you have configured, and ensure
# that the trafficscript!variable_pool_use Global setting is set to 'yes'

$active_sync_pool = "Exchange 2010 ActiveSync Pool";
$owa_pool        = "Exchange 2010 Outlook Web Access Pool";
$oa_pool          = "Exchange 2010 Outlook Anywhere Pool";
$ad_pool          = "Exchange 2010 Auto Discover Pool";

# Please declare the names of the session persistence classes you have created

$universal_session_persistence   = "Exchange Universal Session Persistence";
$transparent_session_persistence = "Exchange Cookie Insert Persistence";

# ------------ end of user-defined parameters

$path = http.getPath();
$outlooksession = http.getCookie( "OutlookSession" );
$auth = http.getHeader( "Authorization" );
$userdata = request.getRemoteIP();
$useragent = http.getHeader( "User-Agent" );
$pool = "";
$sessdata = "";

#Active Sync persistence based on Authorization Header if not persist on Client IP address
if( $path == "/Microsoft-Server-ActiveSync" ) {
  if( $auth ){
    $sessdata = $auth;
  } else {
    $sessdata = $userdata;
  }
  $pool = $active_sync_pool;
}

# Exchange Web Services persistence based on client IP address
else if( string.startsWithI( $path, "/ews" ) ) {
  $sessdata = $userdata;
  $pool = $owa_pool;
}

#Exchange Control Panel persistence based on Transparent Session persistence (Cookie Insert)
else if( string.startsWithI( $path, "/ecp" ) ) {
  $pool = $owa_pool;
}

#Exchange Offline Address Book doesn't need persistence so set $sessdata to "none"
else if( string.startsWithI( $path, "/oab" ) ) {
  $sessdata = "none";
```

$pool = $owa_pool;
}
#Exchange Outlook Anywhere needs persistence based on the client type. Outlook 2010 needs outlooksession cookie else Authorization header
else if( $path == "/rpc/rpcproxy.dll" ) {
    if( string.ContainsI( $useragent, "msrpc" ) ) {
        if( $outlooksession ){
            $sessiondata = $outlooksession;
        } else $sessiondata = $auth;
    } else if( string.ContainsI( $useragent, "microsoft office" ) ) {
        $sessiondata = $auth ;
    }
    $pool = $oa_pool;
}
#Exchange Autodiscover doesn't need persistence so set $sessiondata to "none"
else if( string.startsWithI( $path, "/autodiscover" ) ) {
    $sessiondata = "none"
    $pool = $ad_pool;
}
#Exchange Outlook Web Access needs persistence based on Transparent Session persistence (Cookie Insert) which is default in this rule.
else {
    $pool = $owa_pool;
} pool.select( $pool );

if( $sessiondata != "none" ) {
    if( $sessiondata ) {
        connection.setPersistence( $universal_session_persistence );
        connection.setPersistenceKey( $sessiondata );
    } else {
        connection.setPersistence( $transparent_session_persistence );
    }
}

### Changes in Current Solution Guide

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed the TrafficScript screen shot for Single virtual server for all Exchange Services</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>Updated the TrafficScript for Single Virtual Server for all Exchange services with persistence configuration removed for OAB and AutoDiscover</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### About Riverbed

Riverbed delivers performance for the globally connected enterprise. With Riverbed, enterprises can successfully and intelligently implement strategic initiatives such as virtualization, consolidation, cloud computing, and disaster recovery without fear of compromising performance. By giving enterprises the platform they need to understand, optimize and consolidate their IT,
Riverbed helps enterprises to build a fast, fluid and dynamic IT architecture that aligns with the business needs of the organization. Additional information about Riverbed (NASDAQ: RVBD) is available at www.riverbed.com.