NetScaler and XenMobile Solution for Enterprise Mobility

Deployment Guide

- Load balancing XDMs
- ActiveSync Filtering

Contents

Introduction 3
About This Guide 3
Prerequisites 3
XenMobile XDM 4
XenMobile NetScaler Connector 4
NetScaler Appliance Running Software Version 10.1 4
Deployment Use Case 1: Load Balancing XDM Servers 5
Deployment Topology 5
Configuration Steps Using the Configuration Utility 6
Configuration Steps to Set Up Load Balancing for HTTP Traffic 8
Deployment Use Case 2: Configuring ActiveSync Filtering 8
Deployment Topology 10
Configuration Steps Using the Configuration Utility 11
Troubleshooting Tips 13
Load Balancing Statistics 13
HTTP Callout Statistics 15
Integrated Cache Statistics 17
Appendix 20
Config Summary Use Case 1 20
Config Summary Use Case 2 20
Other References 22

Introduction

Welcome to the NetScaler and XenMobile solution for enterprise mobility deployment guide.

Citrix XenMobile Mobile Device Management (MDM) and Citrix NetScaler® provides a complete, integrated, and scalable solution to the challenges posed by mobility and consumerization of IT around scalability, security, and application visibility.

While XenMobile MDM provides complete protection for your mobile applications, network, and data, and ensures end-to-end security and compliance, NetScaler optimizes, secures, and controls the delivery of all enterprise and cloud services. Together, these two products provide the ability to scale, ensure high availability for apps, and maintain security while reducing mobility deployment and management costs.

NetScaler delivers an extensive portfolio of essential datacenter security capabilities that are significant for mobile users, their apps and data. NetScaler provides critically important application security, network/infrastructure security, and identity and access management, which when combined with XenMobile MDM delivers a tightly coupled solution that enables IT to support the security needs of mobile users and the enterprise.

Deployed directly in front of the servers supporting XenMobile MDM, NetScaler combines high-speed load balancing and content switching, http compression, content caching, SSL acceleration, application flow visibility and a powerful application firewall into a single, easy-to-use platform.

About This Guide

This guide applies to NetScaler appliances running release **10.1** of the NetScaler software and assumes that the required feature licenses are available. This guide provides step-by-step instructions to configure two solutions with XenMobile MDM and NetScaler appliance. The two use cases covered in this guide are:

- Configuring Load balancing and SSL bridging on the NetScaler so that all encrypted traffic is sent directly to the XenMobile Device Manager (XDM) server.
- Configuring ActiveSync filtering by using the XenMobile device management services and the NetScaler.

Prerequisites

Before you begin configuring these deployment scenarios, make sure you have the following already installed:

XenMobile XDM

Use the latest XenMobile XDM. For more information, see http://support.citrix.com/proddocs/topic/cloudgateway/xmob-mdm-landing-page-con.html

XenMobile NetScaler Connector

The XenMobile NetScaler Connector (XNC) provides a device level authorization service of ActiveSync clients to NetScaler acting as a reverse proxy for the Exchange ActiveSync protocol. Authorization is controlled by a combination of policies defined within the XenMobile Device Manager and by rules defined locally by XNC.

XDM provides whitelisting and blacklisting of devices based on compliance with high-level policies such as detection of jailbroken devices or detection of specific apps. The XNC local rules are typically are used to augment the XDM rules in cases where specific overrides are required; for example to block all devices using a specific operating system version.

For more information on XNC, see http://support.citrix.com/proddocs/topic/cloudgateway/xmob-xnc-landing-con.html

NetScaler Appliance Running Software Version 10.1

NetScaler is available as a high-performance network appliance and as software-based virtual appliances for maximum mobile deployment flexibility.

Before you set up a NetScaler MPX or VPX in the datacenter, you need to do the following:

1. Rack mount the MPX appliance, or provision the VPX appliance.

   For instructions

   - about rack mounting a NetScaler MPX appliance, see http://support.citrix.com/proddocs/topic/netscaler-getting-started-map-10/ns-instpk-install-ns-wrapper.html
   - about provisioning a NetScaler VPX virtual appliance, see http://support.citrix.com/proddocs/topic/netscaler-10/ns-gen-nsvpx-wrapper-con-10.html

2. Assign NetScaler IP address (NSIP), subnet mask, and Subnet IP address (SNIP) during initial configuration.

   A NetScaler appliance has both a command line interface (CLI) and a graphical user interface (GUI). The GUI includes a configuration utility for configuring the appliance. For initial access, all NetScaler appliances ship with the default NetScaler IP address (NSIP) of 192.168.100.1 and default subnet mask of 255.255.0.0. However, you can assign a new NSIP and an associated subnet mask during initial configuration.
   For information on configuring the NetScaler appliance for the first time, see

http://support.citrix.com/proddocs/topic/netscaler-getting-started-map-10-1/ns-initial-config-using-ftu-wizard-tsk.html

3. Upload and apply the required license files.
4. Make sure the SSL certificate and private key are available on the appliance or in your local system to use while creating the Load balancing virtual servers.
5. Enable SSL and Integrated Caching features.
    - To enable SSL, access the NetScaler GUI. On the **Configuration** tab, click **Traffic Management**, right-click **SSL Offload**, and then select **Enable**.
    - To enable **Integrated Caching**, on the **Configuration** tab, click **Optimization**, right-click **Integrated Caching**, and then select **Enable**.

Deployment Use Case 1: Load Balancing XDM Servers

The load balancing feature of NetScaler distributes user requests for web pages and other protected applications across multiple servers that all host (or mirror) the same content. You use load balancing primarily to manage user requests to heavily used applications, preventing poor performance and outages and ensuring that users can access your protected applications.

Client devices may connect to the XDM server over ports 80, 443, and 8443. On the NetScaler, you can configure load balancing for secure traffic listening on ports 443 and 8443, as well as configure load balancing for HTTP traffic listening on port 80.

An SSL bridge configured on the NetScaler appliance enables the appliance to bridge all secure traffic directly to the XDM server. The appliance does not offload or accelerate the bridged traffic. The Web server must handle all SSL-related processing. Also, features such as content switching, SureConnect, and cache redirection do not work, because the traffic passing through the NetScaler is encrypted.

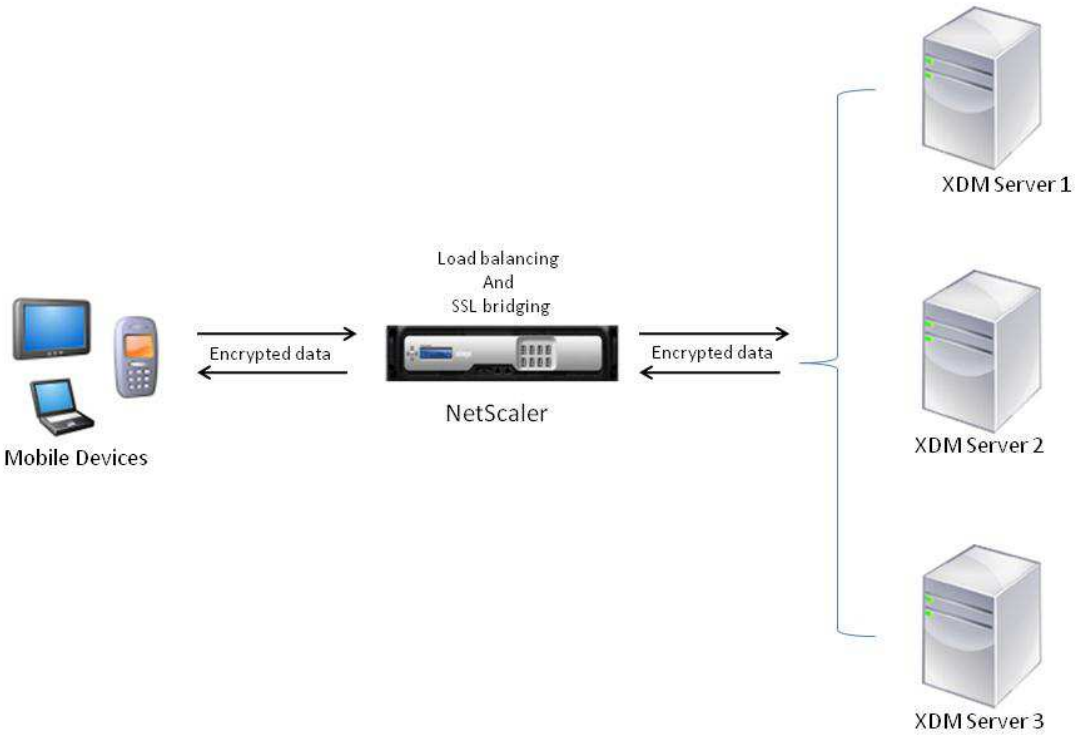
Citrix recommends that you install an acceleration unit (for example, a PCI-based SSL accelerator card) the XDM server to handle the SSL processing overhead.

Features configured in this use case:

- Load balancing
- SSL

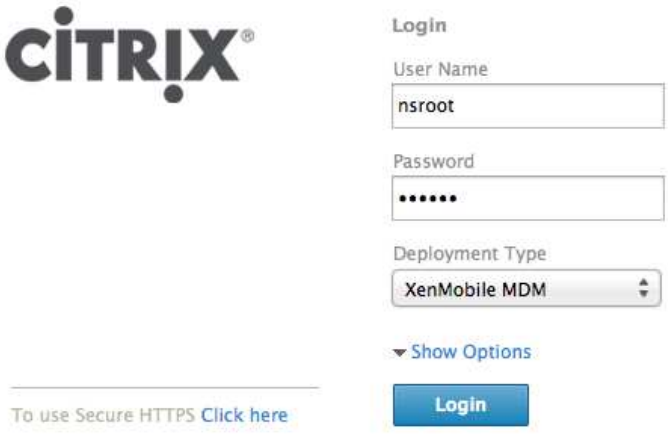
Deployment Topology

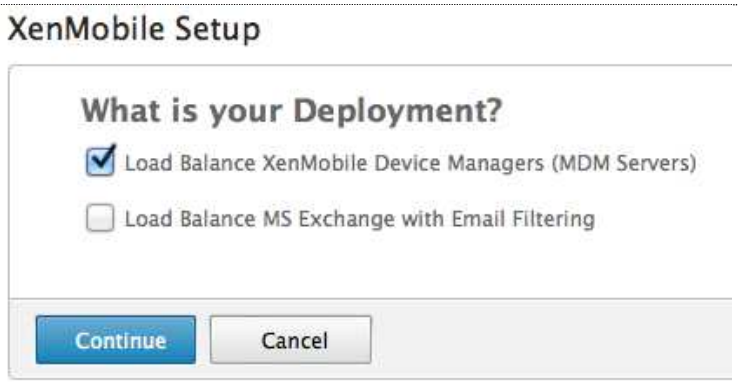
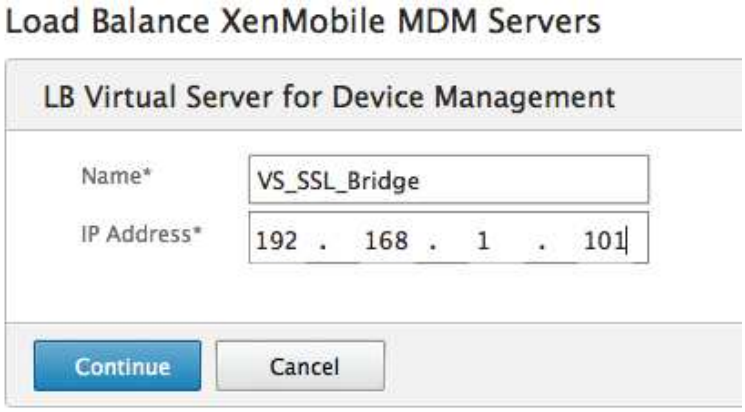
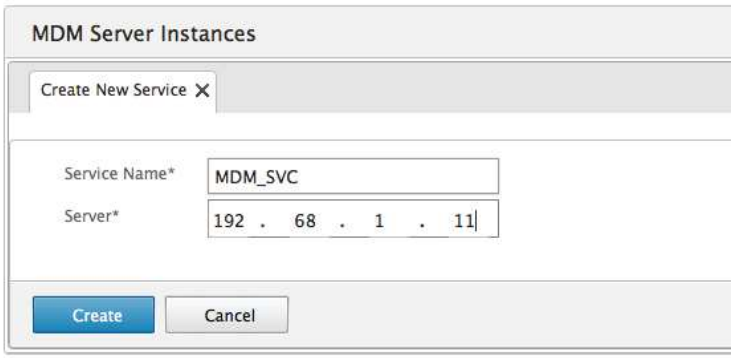
This deployment consists of two key components – NetScaler and XDM servers. The following infographic illustrates the topology where encrypted data from mobile devices is being load balanced to the XDM servers.



Configuration Steps Using the Configuration Utility

Access the configuration utility by typing the appliance's NetScaler IP (NSIP) address in the browser's address bar, and then perform the following steps.

1	On the **Citrix Logon** page, in the **User Name** and **Password** text boxes, type the user name and password of your NetScaler appliance. From the **Deployment Type** drop-down list, select **XenMobile MDM**. From the **Start in** list, select **Configuration**. **Note**: If you have not set the subnet IP address, it will prompt you to do so. Follow the prompts to set SNIP and upload the licenses.	

2	In the **XenMobile Setup** window, select **Load Balance XenMobile Device Managers (MDM Servers)**, and click **Continue**. **Note**: If you have used this wizard earlier to set up load balancing for XenMobile, the **Configuration** tab opens instead of the overview page. In the right pane, click **Load Balance XenMobile Device Managers (MDM Servers)**.	
3	Under **LB Virtual Server for Device Management**, in the **Name** text box, type the name of the load balancing virtual server and in the **IP Address** text box, type the IP address for this virtual server, and then click **Continue**.	
4	Under **MDM Server Instances**, in the **Service Name** text box, type the name of the service to represent the MDM server and in the **Server** text box, type the IP address of the MDM server, and then click **Create**.	

The details of the virtual servers and services created are displayed as shown in the following image.

Configuration Steps to Set Up Load Balancing for HTTP Traffic

In addition to configuring NetScaler for load balancing secure traffic, you can also configure virtual server to load balance HTTP traffic listening on port 80.

1. Perform all the steps mentioned in the previous section Configuration Steps Using the Configuration Utility.
2. On the Configuration tab, in the navigation pane, click **Traffic Management** > **Load Balancing** > **Services**.
3. In the right pane, click **Add** to add a service listening on port 80 for HTTP traffic. In the **Create Service** dialog box, type values in the **Service Name** and **Server** text boxes, from the **Protocol** drop-down list, select HTTP, and in the **Port** text box, type 80. Click **Create**, and then click **Close**.
4. In the navigation pane, click **Load Balancing** > **Virtual Servers**.
5. In the details pane, click **Add** to add a virtual server listening on port 80 for HTTP traffic. In the **Create Virtual Server** dialog box, type values in the **Name** and **IP Address** text boxes. From the **Protocol** drop-down list, select **HTTP** and in the **Port** text box, type 80.
6. On the **Services** tab, select the check box next to the service Svc_HTTP to bind this service to the virtual server. Click **Create**, and then click **Close**.

Deployment Use Case 2: Configuring ActiveSync Filtering

In this scenario, the NetScaler appliance sits between the client and the XNC and CAS servers. All requests from the client devices go to the NetScaler appliance. The NetScaler then sends a request to the XNC with the device details to retrieve information about the device, whether the device is a whitelisted one or a blacklisted one. Based on the response from the XNC, the NetScaler either drops the connection from a blacklisted device or forwards the request from a whitelisted device to the backend server.

The NetScaler also has the capability of storing the callout response from the XNC in the local cache. For subsequent requests from the same device, the NetScaler reuses the stored callout response to make decisions locally to either drop the connection or forward the request.

The following features are configured automatically when you use the XenMobile Setup wizard to configure this use case.

- **Load balancing**: The load balancing feature distributes user requests for web pages and other protected applications across multiple servers that all host (or mirror) the same content. You use load balancing primarily to manage user requests to heavily used applications, preventing poor performance and outages and ensuring that users can access your protected applications. For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-map/ns-lb-wrapper-con-10.html

- **SSL**: A Citrix® NetScaler® appliance configured for SSL acceleration transparently accelerates SSL transactions by offloading SSL processing from the server. For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-map/ns-ssl-wrapper-con-10.html

- **HTTP callout**: For certain types of requests, or when certain criteria are met during policy evaluation, you might want to stall policy evaluation briefly, retrieve information from a server, and then perform a specific action that depends on the information that is retrieved. At other times, when you receive certain types of requests, you might want to update a database or the content hosted on a Web server. HTTP callouts enable you to perform all these tasks.

  An HTTP callout is an HTTP request that the NetScaler appliance generates and sends to an external application when certain criteria are met during policy evaluation. The information that is retrieved from the server can be analyzed by default syntax policy expressions, and an appropriate action can be performed.
  For more information, see http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-map/netscaler-http-callouts-gen-wrapper-10-con.html

- **Responder**: With the **Responder** feature, responses can be based on who sends the request, where it is sent from, and other criteria with security and system management implications. The feature is simple and quick to use. By avoiding the invocation of more complex features, it reduces CPU cycles and time spent in handling requests that do not require complex processing.

  For more information, see http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-map/ns-responder-wrapper-con-10.html

- **Integrated Caching**: The integrated cache feature lets NetScaler store the HTTP callout responses from the XNC in the local cache, and reuse this response for subsequent requests.

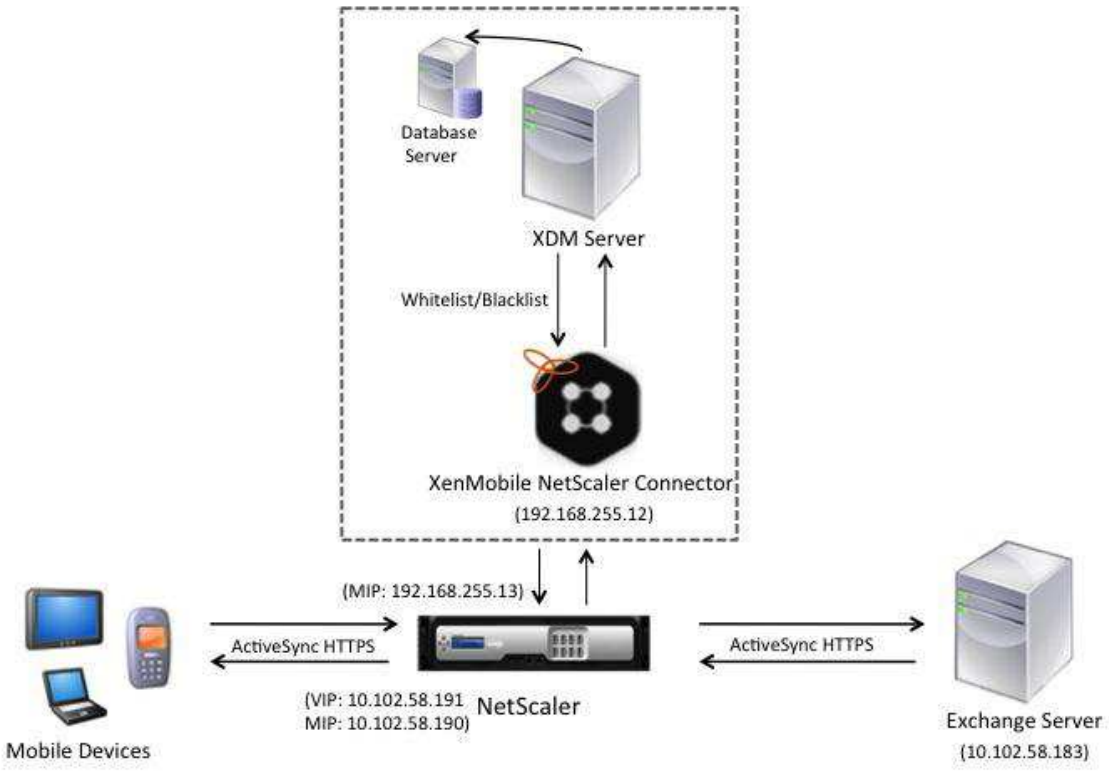
This feature provides in-memory storage on the NetScaler appliance and serves web content to users without requiring a round trip to an origin server. For static content, the integrated cache requires little initial setup. After you enable the integrated cache feature and perform basic setup (for example, determining the amount of NetScaler appliance memory the cache is permitted to use), the integrated cache uses built-in policies to store and serve specific types of static content, including simple Web pages and image files. You can also configure the integrated cache to store and serve dynamic content that is usually marked as non-cacheable by Web and application servers (for example, database records and stock quotes).

When a request or response matches the rule (logical expression) specified in a built-in policy or a policy that you have created, the NetScaler appliance performs the action associated with the policy. By default, all policies store cached objects in and retrieve them from the Default content group, but you can create your own content groups for different types of content.

For more information, see http://support.citrix.com/proddocs/topic/ns-optimization-10-map/ns-IC-gen-wrapper-10-con.html

Deployment Topology

The deployment consists of three key components: – NetScaler appliance, XenMobile Device Manager, and XenMobile NetScaler Connector, as shown in the following infographic.

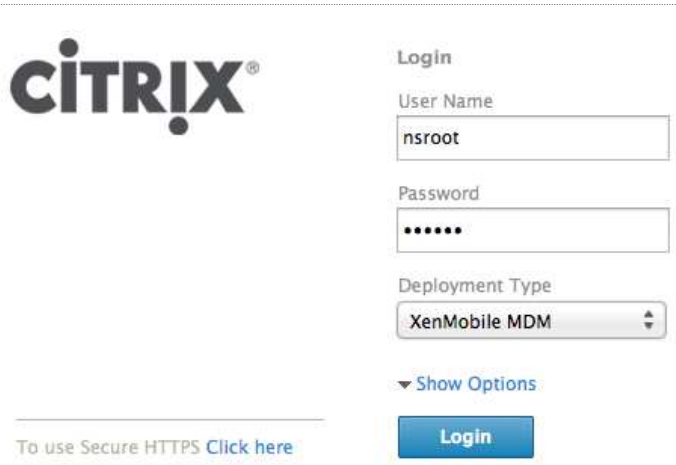
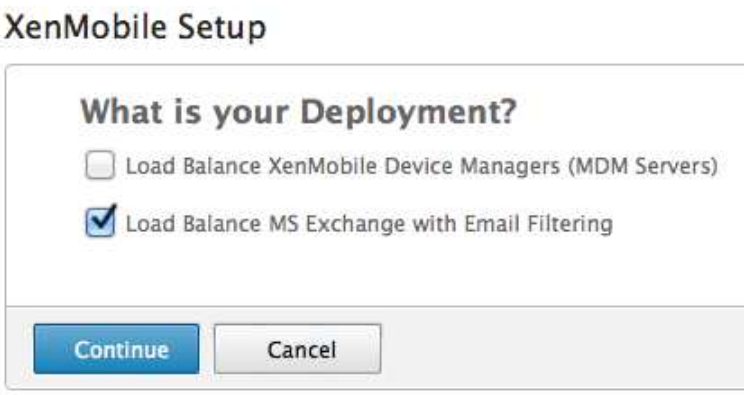


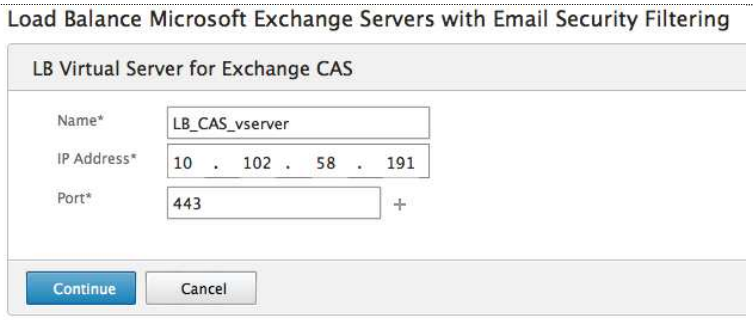
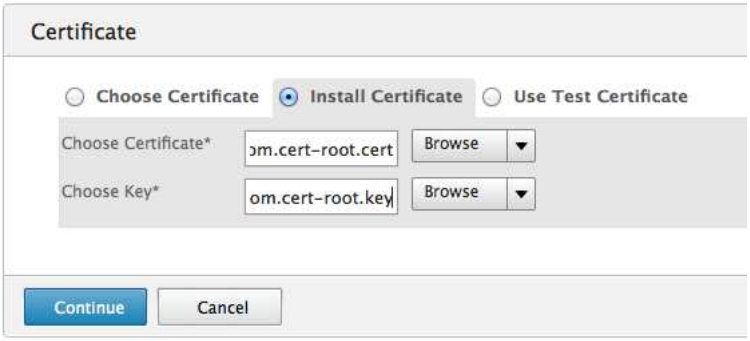
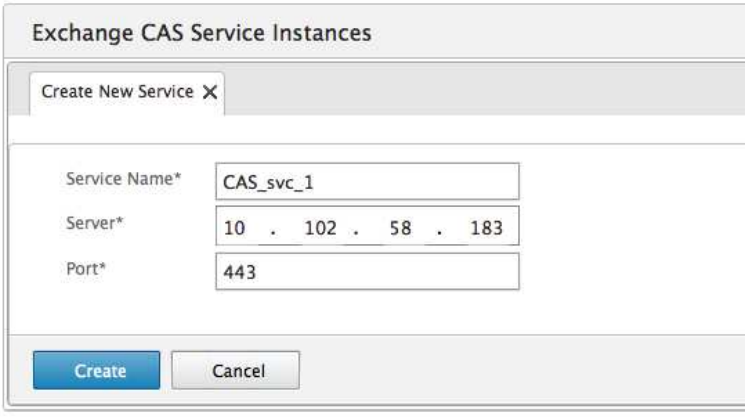
This infographic illustrates the following traffic flow:

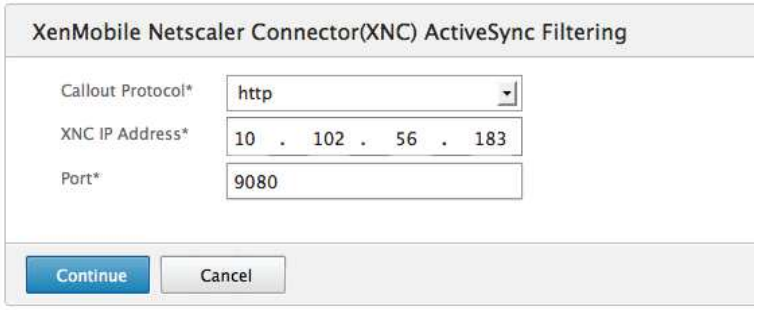
1. First, an ActiveSync request is sent from the client to the NetScaler.
2. Then, the NetScaler sends a request to the XNC server for information on the client device details.
3. Then, the XNC server sends the response - `allow` or `deny` to the NetScaler.
4. If the request is allowed, NetScaler forwards it to the server. If the response is `deny`, NetScaler drops the request.
5. For a request that is allowed, the NetScaler send the server's response to the client.

Configuration Steps Using the Configuration Utility

Access the configuration utility by typing the appliance's NetScaler IP (NSIP) address in the browser's address bar, and then perform the following steps.

1	On the **Citrix Logon** page, in the **User Name** and **Password** text boxes, type the user name and password of your NetScaler appliance. From the **Deployment Type** drop-down list, select **XenMobile MDM**. From the **Start in** list, select **Configuration**. **Note**: If you have not set the subnet IP address, it will prompt you to do so. Follow the prompts to set SNIP and upload the licenses.	
2	In the **XenMobile Setup** window, select **Load Balance MS Exchange with Email Filtering**, and click **Continue**. **Note**: If you have used this wizard earlier to set up load balancing for XenMobile, the **Configuration** tab opens instead of the overview page. In the right pane, click **Load Balance XenMobile Device Managers (MDM Servers)**.	
3	Under **LB Virtual Server for Exchange CAS**, in the **Name**, **IP Address**, and **Port** text boxes, type	

	the required values for the load balancing virtual server, and then click **Continue**.	
4	Under **Certificate**, click **Install Certificate**. In **Choose Certificate**, click **Browse**, and select the certificate file either from the appliance or from your local system, and in **Choose Key**, click **Browse** and select the key file either from the NetScaler appliance or from your local system, and then click **Continue**. **Notes**: Under **Certificate**, if you click **Choose Certificate**, then from the **Certificate** drop-down list, select an already created certificate. Under **Certificate**, if you click **Use Test Certificate**, then in the **Certificate FQDN** text box, type the name of the certificate you want to create.	
5	Under **Exchange CAS Service Instances**, in the **Service Name**, **Server**, and **Port** text boxes, type the values for the service you need to create to represent the CAS server, and then click **Create**.	

6	Click **Create New Service** to add multiple services, and then click **Done**.	
7	Under **XenMobile NetScaler Connector (XNC) ActieSync Filtering**, in **Callout Protocol**, select http. In the **XNC IP Address** text box, type the IP address of your XenMobile NetScaler Connector, and in the **Port** text box, type the port at which the XNC server listens, and then click **Continue**.	
8	You configuration details are displayed. Click **Done** to complete the configuration.	

Troubleshooting Tips

This section provides information to help you troubleshoot issues related to HTTP callout, integrated caching, and SSL offloading.

Load Balancing Statistics

You can view the statistics of the SSL load balancing virtual server by using the following command.

```
stat lb vserver ssl_v1
```

The following output is displayed. The **State** counter shows whether the vserver is in UP state or in DOWN state. For SSL offloading to work, the state should be UP. The **Vserver hits** counter shows you how many hits the virtual server has received.

```
Virtual Server Summary
                vsvrIP        port    Protocol    State   Health
ssl_v1        172.17.0.100   443       SSL        UP      100
Virtual Server Statistics
                                Rate (/s)          Total
Vserver hits                      155              4946
Requests                          155              4946
Responses                         155              4846
Request bytes                   85455202         2508555131
Response bytes                    17453            555673
Total Packets rcvd              115402           3388128
Total Packets sent              115402           3387880
Current client connections        --               100
Current Client Est connections    --               100
Current server connections        --               100
Spill Over Threshold              --               0
Spill Over Hits                   --               0
Labeled Connection                --               0
Push Labeled Connection           --               0
Deferred Request                  0                0
Invalid Request/Response          --               0
Invalid Request/Response Dropped  --               0


Bound Service(s) Summary
                 IP  port       Type      State    Hits     Req
SSLService    192.168.100.3  443        SSL         UP     4946     4946

             Rsp Throughp ClntConn  SurgeQ  SvrConn  ReuseP  MaxConn
SSLService     4846     827     100      0     103       0       0

         ActvTrans  SvrTTFB    Load
SSLService      100     637       0
 Done
```

HTTP Callout Statistics

When the NetScaler appliance encounters an HTTP callout expression, it stalls policy evaluation briefly and sends an HTTP request to the HTTP callout agent by using the parameters configured for the specified HTTP callout. Every request the NetScaler sends to the HTTP callout agent, is recorded by using the **Hits** counter. Use the `show httpcallout` command to view the number of callout hits.

At the NetScaler command prompt, type

```
show httpcallout active_sync_filter
```

The following output is displayed.

Name: active_sync_filter

Vserver: active_sync_filter_vserver (UP)

Return type: TEXT

Method: GET

Host expr: "callout.asfilter.internal"

URL stem: "/services/ActiveSync/Authorize"

Headers: NONE

Parameters: user(HTTP.REQ.HEADER("authorization").AFTER_STR("Basic ").B64DECODE.BEFORE_STR(":")) agent(HTTP.REQ.HEADER("user-agent")) ip(CLIENT.IP.SRC) url(("https://"+HTTP.REQ.HOSTNAME+HTTP.REQ.URL).B64ENCODE) resultType("json")

Result expr: HTTP.RES.BODY(20)

Hits: 13609744

Undef Hits: 0

The **Hits** counter represents the Callout hits.

If the **Hits** counter representing the Callouts Hits does not increment, do one of the following.

1) Verify the network connectivity between clients to the Netscalar by using the Ping command.

2) Verify whether the Vserver counters, represented by **Vserver Hits**, are incrementing while running client traffic.

At the NetScaler command prompt, type

```
stat lb vserver cas_server
```

The following output is displayed. Check the **Vserver Hits** counter.

Virtual Server Summary

	vsvrIP	port	Protocol	State	Health
cas_server	192.168.255.5	443	SSL	UP	100

Virtual Server Statistics

	Rate (/s)	Total
Vserver hits	0	1011
Requests	0	1011
Responses	0	572
Request bytes	0	959105
Response bytes	0	1197905
Total Packets rcvd	0	6141
Total Packets sent	0	5338
Current client connections	--	0
Current Client Est connections	--	0
Current server connections	--	0
Spill Over Threshold	--	0
Spill Over Hits	--	0
Labeled Connection	--	0
Push Labeled Connection	--	0
Deferred Request	0	0
Invalid Request/Response	--	0
Invalid Request/Response Dropped	--	0

Bound Service(s) Summary

	IP	port	Type	State	Hits	Req
cas	172.28.0.127	80	HTTP	UP	1011	1011

	Rsp	Throughp	ClntConn	SurgeQ	SvrConn	ReuseP	MaxConn
cas	572	0	0	0	0	0	0

ActvTrans	SvrTTFB	Load

cas 0 0 0

3) Capture packets on the NetScalar appliance to verify whether the content in the HTTP headers match the Netscalar httpCallout/responder policy and if the XNC receives the Callout responds appropriately. Also, make sure there are no HTTP errors in the response.

   At the NetScaler command prompt, type the following two commands.

```
> start nstrace -size 0
Done
> show nstrace
```

The following output is displayed.

State: RUNNING Scope: LOCAL TraceLocation: "/var/nstrace/25Mar2013_12_11_23/..." Nf: 24 Time: 3600 Size: 0

Mode: TXB NEW_RX Tcpdump: DISABLED PerNIC: DISABLED FileName: 25Mar2013_12_11_23 Link: DISABLED

This shows that trace has started on the NetScaler appliance. Now, send a sample request from the client.

To stop the trace, at the NetScaler command prompt, type

```
> stop nstrace
 Done
```

To enter the NetScaler shell prompt, at the NetScaler command prompt, type

```
> shell
```

Then, use the `cd` command to navigate to the `/var/nstrace/<timestamp>/` directory. You will find a file named `nstrace1.cap`.

Download the captured pcap, by using SCP or FTP to any computer and open the file using Wireshark.

Integrated Cache Statistics

To ensure that the caching feature on the NetScaler appliance works properly, you need to perform the following tasks.

1. Make sure that you have configured Network Time Protocol (NTP) servers and enabled NTP synchronization on both the XNC server and the NetScaler appliance. NTP configuration is required for caching to work properly.

   You can configure your NetScaler appliance to synchronize its local clock with an NTP server. This ensures that its clock has the same date and time settings as the other servers on your network.

   You can configure clock synchronization on your appliance by adding NTP server entries to the ntp.conf file from either the configuration utility or the NetScaler command line, or by manually modifying the ntp.conf file, and then starting the NTP daemon (NTPD). The clock synchronization configuration does not change if the appliance is restarted, upgraded, or downgraded. However, the configuration does not get propagated to the secondary NetScaler in a high availability setup.

   For instructions on setting up the NTP server, see http://support.citrix.com/proddocs/topic/ns-system-10-map/ns-ac-config-clk-sync-using-conutil-cli-tsk.html

   Note: If you do not have a local NTP server, you can find a list of public, open access, NTP servers at the official NTP site, http://www.ntp.org, under Public Time Servers List. Before configuring your NetScaler to use a public NTP server, be sure to read the Rules of Engagement page (link included on all Public Time Servers pages).

2. You should allocate memory to the cache for storing objects. You can verify this by using the `show cache parameter` command.

   At the NetScaler command prompt, type

   ```
   show cache parameter
   ```

   The following line should appear in the output.

   Memory usage limit (active value): <memory set> Mbytes

   Example:

   > show cache parameter

   Integrated cache global configuration:
   Memory usage limit: 6000 MBytes
   Memory usage limit (active value): 4000 MBytes
   Maximum value for Memory usage limit: 7824 MBytes
   Via header: NS-CACHE-9.2: 230
   Verify cached object using: HOSTNAME
   Max POST body size to accumulate: 4096 bytes
   Current outstanding prefetches: 0

Max outstanding prefetches: 4294967295
Treat NOCACHE policies as BYPASS policies: YES
Global Undef Action: NOCACHE

If this value does not appear in the output, it implies that memory has not been allocated to the cache. If you do not allocate memory to integrated cache, all requests are sent to the server. Therefore, you need to allocate memory to the cache by using the `set cache parameter` command. For any model of the NetScaler appliance, you can allocate half of the memory to the cache. However, Citrix recommends allocating a little less than half of the memory, because of internal memory dependency. You can run the following command at the NetScaler command prompt to allocate 512 MB of memory to cache.

```
set cache parameter –memLimit 512
```

You can use either of the following options to display the statistics for cache:

- **`stat cache`**
  To display the summary of the cache statistics.

- **`stat cache –detail`**
  To display the full details of the cache statistics.

In the output, the **Hits** counter gives you the total number of responses served by the cache.
Example:

```
> stat cache
```

Integrated Cache Statistics - Summary

	Rate (/s)	Total
Hits	158	13754735
Misses	158	13755169
Requests	317	27509904
Hit ratio(%)	--	50
Origin bandwidth saved(%)	--	0
Cached objects	--	1
Marker objects	--	0
Hits being served		0
Misses being handled		0

Appendix

The appendix provides you with a summary of the configurations in the two use cases in this deployment guide.

Note: All configurations have been tested on NetScaler software version 10.1.118.7.

Config Summary Use Case 1

- Load balancing virtual server
- XDM Server

Config example:

```
enable ns feature lb
add ns ip 192.168.1.1 255.255.255.0 -type MIP -vServer DISABLED

add service Svc_SSL_Bridge_443 192.68.1.11 SSL_BRIDGE 443
add service Svc_SSL_Bridge_8443 192.68.1.11 SSL_BRIDGE 8443
add service Svc_HTTP 192.68.1.11 HTTP 80

add lb vserver VS_SSL_Bridge_443  SSL_BRIDGE 192.168.1.101 443
add lb vserver VS_SSL_Bridge_8443  SSL_BRIDGE 192.168.1.101 8443
add lb vserver VS_HTTP HTTP 192.168.1.101 80

bind lb vserver VS_SSL_Bridge_443 Svc_SSL_Bridge_443
bind lb vserver VS_SSL_Bridge_8443 Svc_SSL_Bridge_8443
bind lb vserver VS_HTTP Svc_HTTP
```

Config Summary Use Case 2

- XNC server – HTTP Callout
- ActiveSync service
- SSL load balancing virtual server
- Caching
- Responder

Config example:

```
enable ns feature LB SSL IC RESPONDER
add ns ip 10.102.58.190 255.255.255.128 -type MIP

add service XNC1 10.102.58.200 HTTP 9080
add lb vserver active_sync_filter_vserver HTTP 0.0.0.0 0
bind lb vserver active_sync_filter_vserver XNC1
```

```
   add service CAS1 10.102.58.183 SSL 443
   add lb vserver CASVserver SSL 10.102.58.191 443
   bind lb vserver CASVserver CAS1

   add ssl certKey pi-srv -cert "/nsconfig/ssl/pi-srv.cert" -key
"/nsconfig/ssl/pi-srv.key"
   bind ssl vserver CASVserver -certkeyName pi-srv

   add policy httpCallout active_sync_filter
   add policy httpCallout active_sync_filter_deviceid
   set policy httpCallout active_sync_filter -vServer
active_sync_filter_vserver -returnType TEXT -hostExpr
"\"callout.asfilter.internal\"" -urlStemExpr
"\"/services/ActiveSync/Authorize\"" -parameters
user(HTTP.REQ.HEADER("authorization").AFTER_STR("Basic
").B64DECODE.BEFORE_STR(":").HTTP_URL_SAFE)
agent(HTTP.REQ.HEADER("user-agent").HTTP_URL_SAFE) ip(CLIENT.IP.SRC)
url(("https://"+HTTP.REQ.HOSTNAME+HTTP.REQ.URL).B64ENCODE)
resultType("json") -resultExpr "HTTP.RES.BODY(20)"
   set policy httpCallout active_sync_filter_deviceid -vServer
active_sync_filter_vserver -returnType TEXT -hostExpr
"\"callout.asfilter.internal\"" -urlStemExpr
"\"/services/ActiveSync/Authorize\"" -parameters
user(HTTP.REQ.HEADER("authorization").AFTER_STR("Basic
").B64DECODE.BEFORE_STR(":").HTTP_URL_SAFE)
agent(HTTP.REQ.HEADER("user-agent").HTTP_URL_SAFE) ip(CLIENT.IP.SRC)
url(("https://"+HTTP.REQ.HOSTNAME+HTTP.REQ.URL).B64ENCODE)
resultType("json") DeviceId(HTTP.REQ.URL.QUERY.VALUE("DeviceId")) -
resultExpr "HTTP.RES.BODY(20)"

   add responder policy active_sync_filter
"HTTP.REQ.URL.QUERY.CONTAINS(\"DeviceId\").NOT &&
HTTP.REQ.URL.STARTSWITH(\"/Microsoft-Server-ActiveSync\") &&
HTTP.REQ.METHOD.EQ(POST) &&
HTTP.REQ.HOSTNAME.EQ(\"callout.asfilter.internal\").NOT &&
SYS.HTTP_CALLOUT(active_sync_filter).SET_TEXT_MODE(IGNORECASE).CONTA
INS(\"allow\").NOT" DROP
   add responder policy active_sync_filter_deviceid
"HTTP.REQ.URL.QUERY.CONTAINS(\"DeviceId\") &&
HTTP.REQ.URL.STARTSWITH(\"/Microsoft-Server-ActiveSync\") &&
HTTP.REQ.METHOD.EQ(POST) &&
HTTP.REQ.HOSTNAME.EQ(\"callout.asfilter.internal\").NOT &&
SYS.HTTP_CALLOUT(active_sync_filter_deviceid).SET_TEXT_MODE(IGNORECA
SE).CONTAINS(\"allow\").NOT" DROP

   set cache parameter -memLimit 200 -via "NS-CACHE-10.0: 180"
   add cache selector Url_Match "HTTP.REQ.URL.QUERY.VALUE(\"url\")"
```

```
   add cache selector DeviceId_Match HTTP.REQ.URL.PATH
HTTP.REQ.HOSTNAME "HTTP.REQ.URL.QUERY.VALUE(\"DeviceId\") +  \"-\" +
HTTP.REQ.URL.QUERY.VALUE(\"user\")"
   add cache contentGroup Req_with_DeviceId -relExpiry 300 -
hitSelector DeviceId_Match
   add cache contentGroup Req_without_DeviceId -relExpiry 300 -
hitSelector Url_Match

   add cache policy cache_req_with_DeviceId -rule
"HTTP.REQ.HEADER(\"Host\").CONTAINS(\"callout\") &&
HTTP.REQ.URL.QUERY.CONTAINS(\"DeviceId\")" -action CACHE -
storeInGroup Req_with_DeviceId
   add cache policy cache_req_without_DeviceId -rule
"HTTP.REQ.HEADER(\"Host\").CONTAINS(\"callout\") &&
HTTP.REQ.URL.QUERY.CONTAINS(\"DeviceId\").NOT &&
HTTP.REQ.URL.QUERY.CONTAINS(\"url\")" -action CACHE -storeInGroup
Req_without_DeviceId

   bind lb vserver active_sync_filter_vserver -policyName
cache_req_without_DeviceId -priority 90 -gotoPriorityExpression END
-type REQUEST
   bind lb vserver active_sync_filter_vserver -policyName
cache_req_with_DeviceId -priority 100 -gotoPriorityExpression END -
type REQUEST

   bind lb vserver CASVserver -policyName
active_sync_filter_deviceid -priority 90 -gotoPriorityExpression END
-type REQUEST
   bind lb vserver CASVserver -policyName active_sync_filter -
priority 100 -gotoPriorityExpression END -type REQUEST
```

Other References

- Scaling and Capacity Guidelines

  See internal Sales Knowledge Base for these guidelines
- XenMobile and NetScaler Whitepaper

  http://citrix.com/content/dam/citrix/en_us/documents/products/deliver_enterprise_mobility_xenmobile_netscaler.pdf

- Use case 1 blog:

  http://blogs.citrix.com/2013/03/12/fronting-xenmobile-mdm-with-netscaler/

- SSL Bridging:

http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-map/ns-ssl-bridging-tsk.html

- HTTP Callout:

  http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-map/netscaler-http-callouts-gen-wrapper-10-con.html