

NetScaler Deployment Guide

Replacing Microsoft Forefront TMG with Citrix NetScaler for Website Publishing

Authors

The following authors contributed to the creation of this deliverable.

Citrix

Abhishek RVRK Sharma
#33, Ulsoor Road
Bangalore, Karnataka 560042
India
Phone: +91 80 39541000 Extension 78219
Abhishek.sharma@citrix.com

Revision History

Revision	Change Description	Updated By	Date
1	Initial Draft	Abhishek RVRK Sharma	6 th May 2015

Table of contents

Introduction.....	4
Configuration details	5
Solution description	7
Section 1 – Define IPs and addressing for the load balanced websites	7
Section 2 - Website Switching Policy	11
Section 3 – Security definition	14
Additional information	19
SSL bridging and tunneling	19
Authentication	24
Conclusion.....	26

Introduction

This deployment guide defines a solution for replacing Microsoft Forefront Threat Management Gateway (TMG) with Citrix® NetScaler®, a world-class application delivery controller (ADC) with the proven ability to load balance, accelerate, optimize and secure enterprise applications.

Forefront TMG is a network firewall program with attack filtering, VPN and basic application caching and load balancing capabilities. It runs on Windows Server 2008 and works by inspecting all network traffic that passes through it.

Forefront TMG offers several useful network protection features:

Routing and remote access features: It can act as a router, an Internet gateway, a virtual private network (VPN) server, a network address translation (NAT) server or a proxy server.

Security features: It is a firewall capable of inspecting network traffic (including web content, secure web content and email) to filter out malware, identify and stop security vulnerability exploit attempts and provide content filtering according to predefined security policies. Therefore, Forefront TMG can provide application layer protection, stateful and content filtering and anti-malware protection.

Network performance features: Forefront TMG can help improve network performance with web traffic compression and web caching, which allows frequently accessed web resources to be cached so they can be accessed faster. Microsoft Forefront TMG 2010 can also cache data received through Background Intelligent Transfer Service (BITS), such as Microsoft Updates.

As an enterprise networking product, Forefront TMG is used by a large number of small and medium businesses as well as large enterprises. However, on September 9, 2012, Microsoft announced that no further development would take place, and that the product would no longer be available for purchase as of December 1, 2012. Mainstream support ceased in April 2015 and extended support will end on April 14, 2020. Because Microsoft is retiring the product, enterprises must replace it. This guide presents the capabilities of NetScaler as a robust and enhanced platform to replace Forefront TMG where it has been deployed as a reverse proxy for publishing a single website or multiple sites in a load balanced setup.

The following TMG features are replicated with NetScaler -

Secure Application Publishing

- Secure publishing of web and internal servers
- Single sign-on (SSO)
- Delegation of basic authentication
- Link translation
- SSL bridging
- SSL tunneling

Firewall Protection

- Multi-layer firewall
- Application-layer filtering
- Granular HTTP controls
- Denial of Service (DoS) protection
- Extensive protocol support

Networking and Performance

- Network load balancing
- Network-based configuration
- Caching
- HTTP compression
- QoS (using Diffserv)

Configuration details

To show Forefront TMG administrators the equivalent steps for configuring NetScaler, this guide presents side-by-side, step-by-step details for a sample use case involving Microsoft Exchange 2013, depicted in Figure 1.

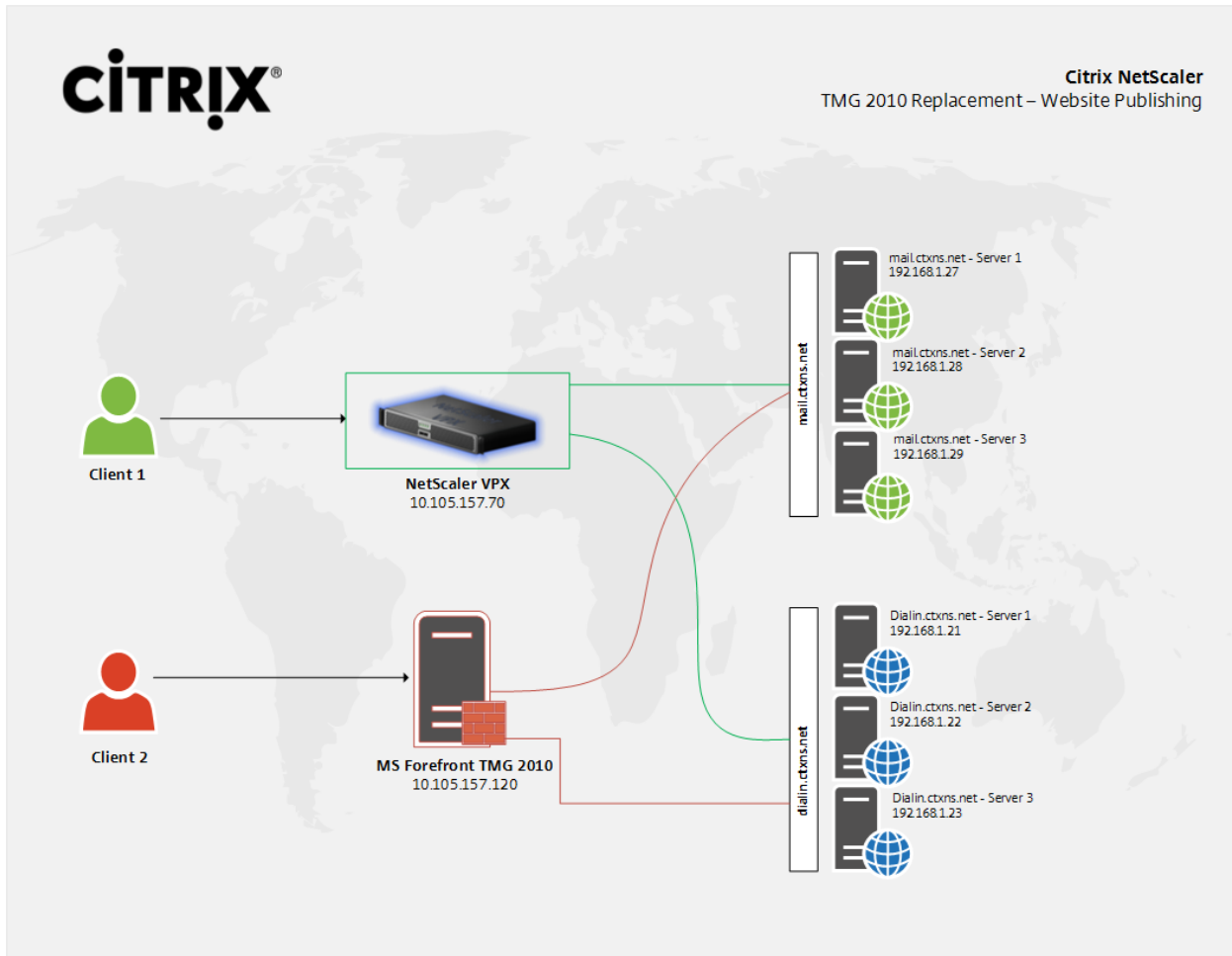


Figure 1. Diagram of test setup

Product	Version
Forefront TMG	2010 SP1
NetScaler VPX™	10.5 (enterprise license)

NetScaler features to be enabled

The following NetScaler features are used in this Exchange 2013 deployment. Please ensure they are enabled in the NetScaler system.

- Content switching
- Load balancing
- SSL offload

Here is a quick explanation of how these features work.

Content switching - The content switching module directs incoming traffic to a matching load balancing virtual server. This logical switching of incoming traffic based on content type allows configuration of specific optimization policies.

Load balancing - NetScaler load balancing evenly distributes requests to backend servers. Multiple algorithms (such as LEASTCONNECTION, ROUNDROBIN, etc.) are supported to provide efficient load balancing logic for every application server.

SSL offload - SSL connections are terminated at the NetScaler appliance, thus allowing NetScaler to conduct advanced traffic monitoring. Additionally, SSL offload can significantly reduce the computational overhead of offloading encrypted user connections on backend servers.

For enabling additional capabilities, authentication, authorization and auditing (AAA), rewrite, responder, HTTP compression, DoS protection, clustering, and integrated caching could also be enabled. However, the use case described here can be deployed using the features described above.

Solution description

Forefront TMG

NetScaler VPX

Section 1 – Define IPs and addressing for the load balanced websites

Step 1 – Initialize the Web Publishing Rule Wizard

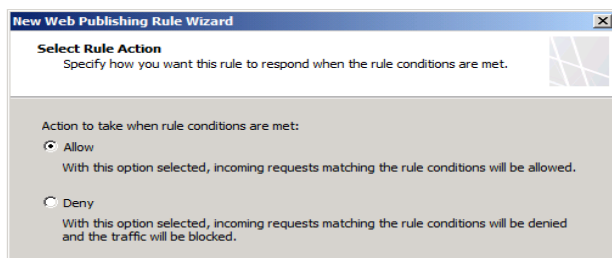
First, start Forefront TMG and get to the console. Then right-click on the **Firewall Policy** node in the left pane of the console. Then, select **New > Website Publishing Rule**.

Alternatively, after clicking on **Firewall Policy** in the left pane, you will see new tasks populated in the Tasks Tab on the Task Pane on the right side of the TMG firewall console. Here, under **Firewall Policy Tasks**, click on the **Publish Web Sites** link to start the Web Publishing Rule Wizard.

This will bring up the **Welcome to the New Web Publishing Rule Wizard** page. Enter an appropriate name for the rule in the text box provided here. This name will be used in the list of firewall policy rules, so the name should be identifiable.

Step 2 – Select the Firewall Rule action

This leads to the **Select Rule Action** page, where the choice of whether to make this an allow or deny rule will be presented. Here, we choose the **Allow** option as we are looking to allow traffic through to the websites. Deny is a useful option when user access to a website needs to be restricted.



Step 3

The next page, the **Publishing Type** page, provides a set of choices: publish a single site,

Step 1- Define the load balancing virtual servers (LB vservers)

First, log into the NetScaler GUI. On the Configuration tab, move to **Traffic Management>Load Balancing>Virtual Servers**.

For this deployment exercise, we have two websites, one hosting Lync and the other hosting Outlook Web Access, hence we have defined the two load balancing virtual servers as **LBVS_TMG_Lync**, and **LBVS_TMG_OWA**. (shown in a later screenshot)

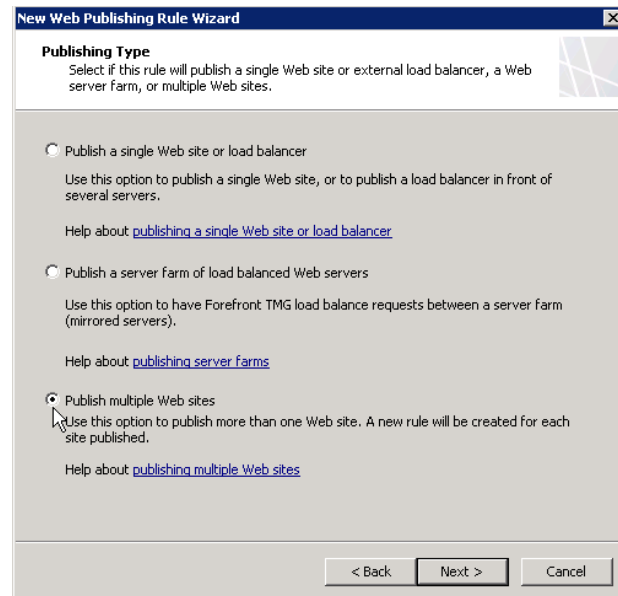
Alternatively, if you have more than two or only a single website, you can create the corresponding number of LB vservers and bind all of them to the content switching vserver (along with individual content switching policies for each website) that will serve as the reverse proxy for all of these websites.

Step 2 - Configure the LB vservers

When defining a new LB vserver, you will be presented with the settings screen. Here, enter the settings appropriate for your setup.

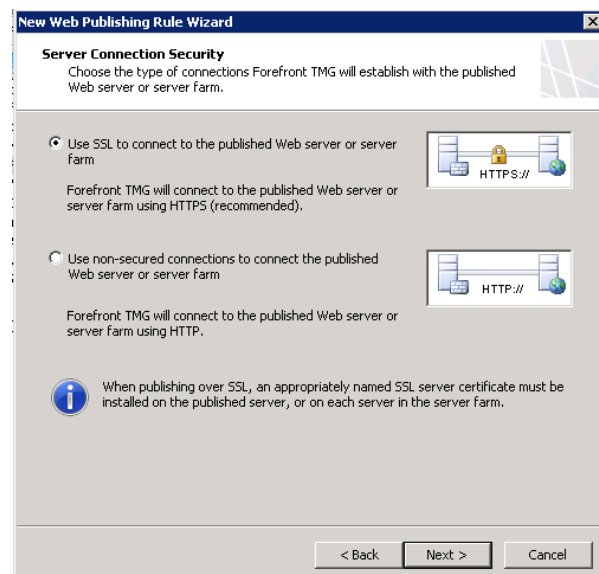
- Note that the protocol here is presented as **HTTP**. You can choose **SSL** as well, in which case you would be required to provide a valid certificate for this server.
- Set the IP address type as **Non Addressable**. This is because the address will be set upon the content switching virtual server (CS vserver) that we will define later.
- Leave the other settings as is.

publish multiple sites or publish a web farm. If you choose to publish a web farm, the TMG firewall will perform basic, round-robin load balancing for you, which while effective is limiting and not very viable for high-performance enterprise web applications.

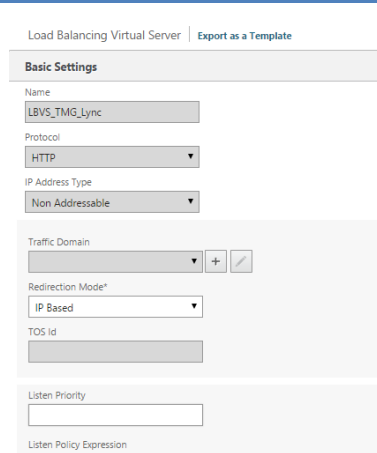


In this example, we're going to publish multiple websites so we will select the **Publish multiple websites** option. Alternatively, you may choose to publish a single website as well using the **Publish a single website or load balancer** option. Click **Next**.

Step 4 – Server connection security



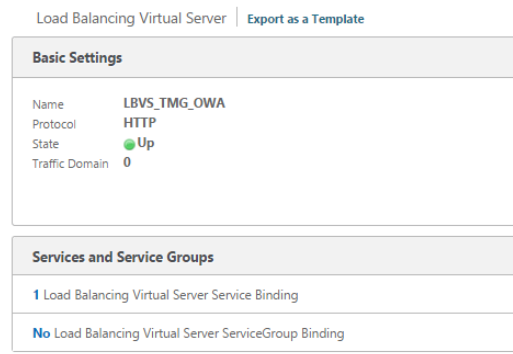
On the Server Connection Security page, you are presented the option to connect securely to the published web server. If this option is chosen, an SSL connection will be established between the



After clicking **OK**, move back to the configuration screen for the LB vserver. **Enable a useful session persistence method (such as SSLSESSION)** to ensure user sessions are maintained during load balancing.

NetScaler advantage

TMG supports only two session persistence options: cookie based and source IP based. However, NetScaler supports 10 different persistence methods, including an option to create a user -defined rule for persistence, thus making it possible to have an unlimited set of combinations to identify connections that must be part of the same persistence session.

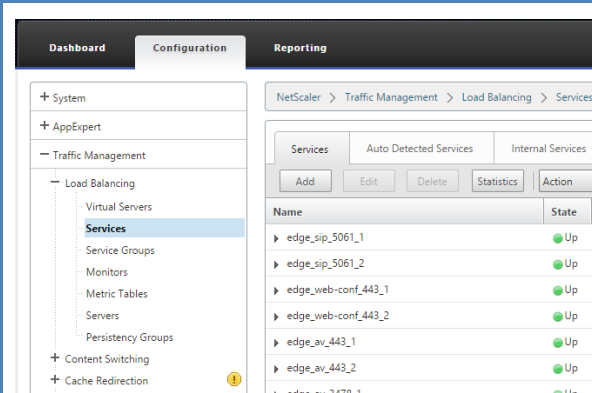


Now click on the **Load Balancing Virtual Server Service Binding** tab in the **Service and Service Groups** section as shown above, or alternatively, click on **Services** in the **Traffic Management>Load Balancing** subsection and then, click on the **Add** button (shown below).

TMG firewall and the published server. The published server must have an SSL certificate installed that is trusted by TMG.

Step 5 – Website definition

The steps here will vary depending on whether you choose to load balance your website or not. Since our current use case defines a load balanced website, we will proceed as such.



Step 3 – Define LBVS server service binding

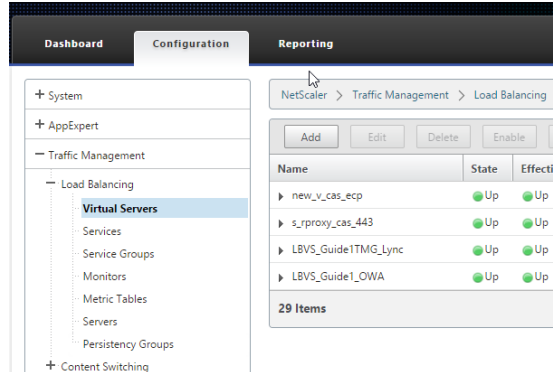
Every LB service is linked to a server; this can either be a new server or an existing server already defined in the **Servers** subsection under **Load Balancing**.

The screenshot shows the 'Load Balancing Service' configuration form. The 'Basic Settings' section is active. The 'Service Name*' field contains 's_rproxy_4443_director_vip'. The 'New Server' radio button is selected. The 'IP Address*' field is empty, with an 'IPv6' checkbox. The 'Protocol*' dropdown is set to 'SSL'. The 'Port*' field contains '4443'. The 'Traffic Domain' dropdown is empty. The 'Hash ID' field is empty. The 'Server ID' dropdown is set to 'None'. The 'Clear Text Port' field is empty. The 'Cache Type*' dropdown is set to 'SERVER'. The 'Cacheable' checkbox is unchecked. The 'Enable Service', 'Health Monitoring', and 'AppFlow Logging' checkboxes are checked. The 'Number of Active Connections' field contains '0'. The 'Comments' field is empty. At the bottom, there are 'OK' and 'Cancel' buttons.

Here, define the name for the service (**s_rproxy_4443_director_vip for this deployment**), the IP address (or choose from a list in the case of an existing server) for the new server and the protocol it operates on. For this deployment, the IPs will correspond to 192.168.1.27, 192.168.1.28, 192.168.1.29 for the first server and 192.168.1.21, 192.168.1.22 and 192.168.1.23 for the second one.

You may enable **Health Monitoring** if you would like to have NetScaler poll the server periodically to verify its health. If Health Monitoring is disabled, the appliance shows the server **UP** at all times.

Finally, the LB vservers created will be displayed on the configuration screen as shown below.



NetScaler advantage

TMG does not allow you to run this wizard for multiple websites that can each run as a server farm, that is, multiple load balanced websites. In order to enable this, you will need to run this rule once each for each website that needs to be load balanced (using the Publish a server farm of load balanced web servers option), and then run it once again with the Publish multiple websites rule. When running this rule, you have to specify the website names that you have defined in the rules created earlier for each load balanced website.

This process is very simple with NetScaler, however. To enable this use case, all you will need to do is to add internal server definitions to the LB vserver. NetScaler also currently supports 15 different load balancing techniques (ROUNDROBIN, LEASTCONNECTION, LEAST RESPONSE TIME and hashing based techniques such as URLHASH, DOMAINHASH, etc.), thus allowing great flexibility for load balancing definition.

Section 2 - Website Switching Policy

Step 5a – Specify server farm to publish

New Web Publishing Rule Wizard

Specify Server Farm
Specify the Web server farm that you would like to publish

Select the server farm you want to publish:

Edit...
New...

Select how Forefront TMG will load balance incoming Web requests:

Cookie-based Load Balancing
 Source-IP based Load Balancing

Help about [load balancing mechanisms](#)

< Back Next > Cancel

You can either choose to select a pre-existing server farm (If already defined in TMG) or create a new one. Forefront TMG will load balance all requests using a round robin algorithm. The load balance selector here is misleading, as the two options (**Cookie-based and Source-IP based Load Balancing**) indicate session persistence options.

Step 5b – Create a new server farm

Clicking **New** on the earlier prompt brings up the **New Server Farm wizard**. Give a name to the new farm and click on **Next**.

Step 4 – Define content switching virtual server (CS vserver)

Content Switching Virtual Server

Basic Settings

Name*
TMG_CSVS_1

Protocol*
SSL

IP Address Type*
IP Address

IP Address*
10 . 105 . 157 . 129 IPv6

Port*
443

Traffic Domain
 +

Range
1

Listen Priority

Listen Policy Expression
Operators Saved Policy Expressions
Press Control+Space to start the expression and then type

Comments

State
 RHI State
 AppFlow Logging

Less

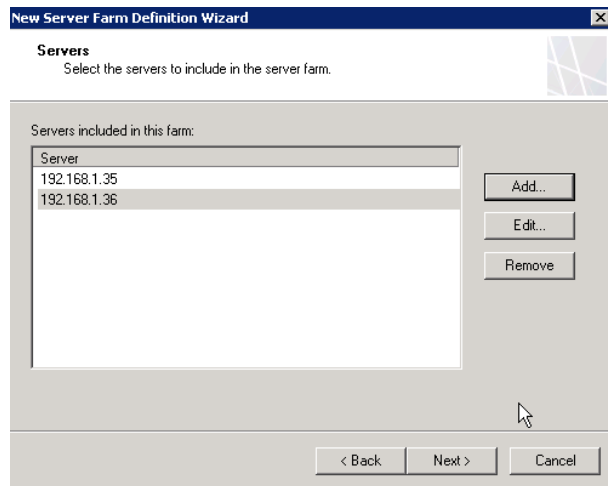
OK Cancel

Now, define the CS vserver. To get to the screen above, move to **Traffic Management>Content Switching>Virtual Servers** and click on the **Add** button.

For this deployment, the SSL protocol has been used, as the websites will be accessed over HTTPS. You can, alternatively, choose to use HTTP.

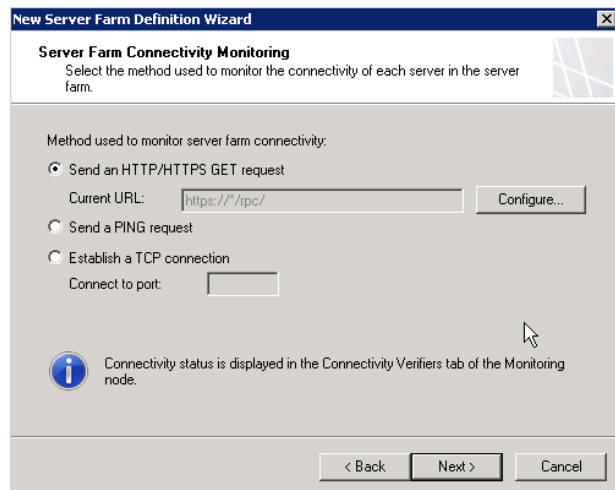
Note: If you have chosen to define your CS vservers as SSL, then your LB vservers (defined earlier) cannot be SSL. Also, you cannot use Session ID as the persistence method.

Step 5c- Server definition



Next, you can define the servers that make up this farm. Essentially, here you may define the IPs of the servers that are to be used to load balance this website.

Step 5d – Server farm connectivity monitoring



Here, you can define the method used to monitor the health of the servers forming the server farm (corresponds to Health Monitoring on NetScaler).

For IP address type, you can use either **IP Address** or **IP Pattern**. IP Pattern will link the CS vserver to a pattern-based set of IP addresses, useful when the virtual server has to respond to requests coming to multiple IPs. Define the IP address for the vserver; the port will be automatically set to port 443 for SSL, but if you are using an alternate port you may specify the same.

Leave the other settings as is, and click **OK**.

Step 5 – CS policy definition

Content Switching Virtual Server

Basic Settings

Name **TMG_CSVS_1**
Protocol **SSL**
State **UP**
IP Address **10.105.157.129**
Port **443**

CS Policy Binding

2 Content Switching Policies

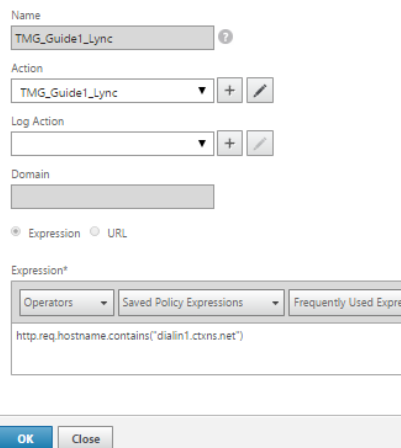
No Default Load Balancing Virtual Server Bound

ECC Curve

4 ECC Curves

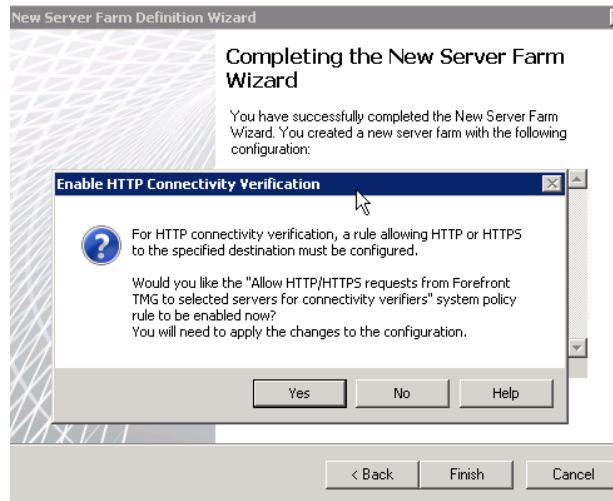
Click on the name of the CS vserver you have just created in the list on the main panel at **Traffic Management>Content Switching>Virtual Servers** and click on **CS Policy Binding>Content Switching Policies**.

Configure Content Switching Policy



Configure the policy as shown in the screen above. You will need to create two policies, one

Step 5e –Complete server farm creation

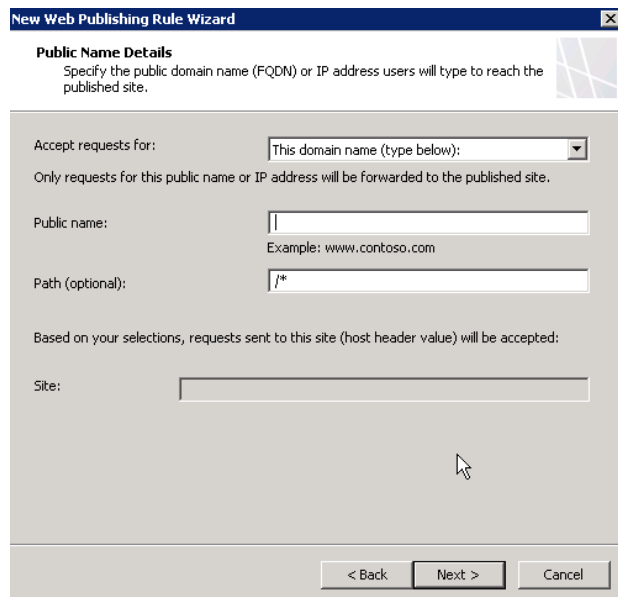


After completing the wizard, you may be prompted with a request to enable a system policy for HTTP connectivity verification to enable monitoring.

Step 6 – Public name/FQDN definition

This step will present itself after the completion of Step 6e for load balanced websites, or immediately after Step 5 in the case of multiple websites. As stated earlier, both these wizards will have to be run independently for multiple load balanced websites.

For server farms, this is the screen that will be presented. As you will note, this screen will show the host header value that will be accepted for this farm.



for each website that you need to serve. Here, we have two policies, with the expression:

```
http.req.hostname.contains("<website domain name>")
```

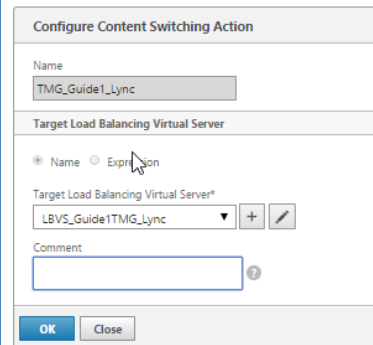
For the deployment, the two policies use the following expressions:

```
http.req.hostname.contains("mail.ctxns.net")
```

```
http.req.hostname.contains("dialin.ctxns.net")
```

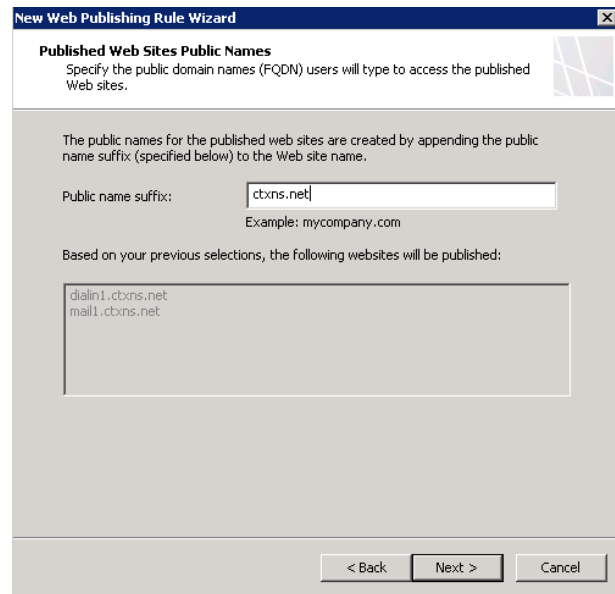
Click on the icons next to the **Action** header to either add a new action or edit an existing action.

Step 6 – CS action definition



When presented with the screen above, define the appropriate load balancing virtual server and click **OK**.

And for multiple websites, this is the screen presented (you will notice that the public names for all individual websites are added here).

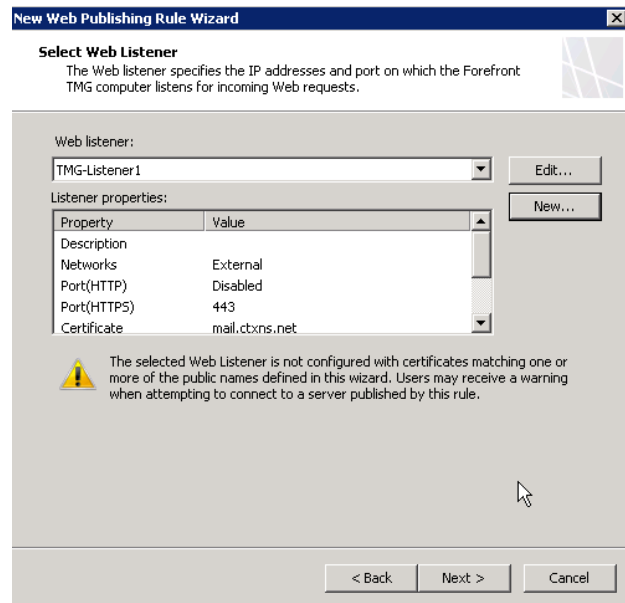


Section 3 – Security definition

Step 7 – Define web listener

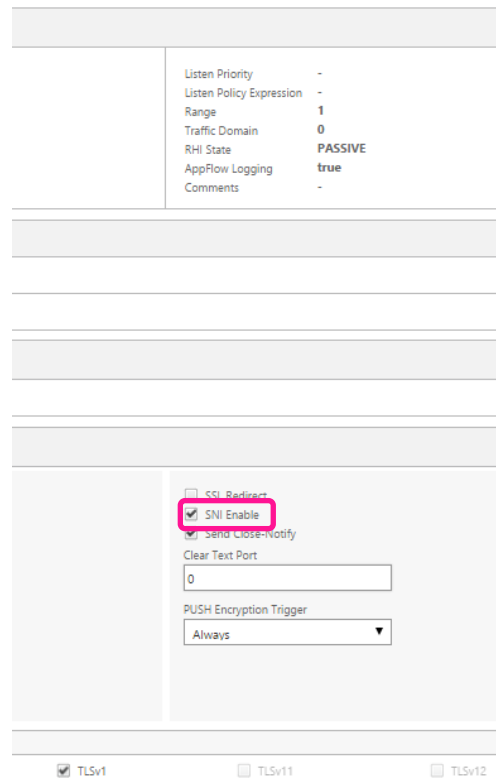
The next screen will require you either to identify an existing web listener or to create a new listener to be used with this website.

If you already have a web listener defined, this is what is shown:

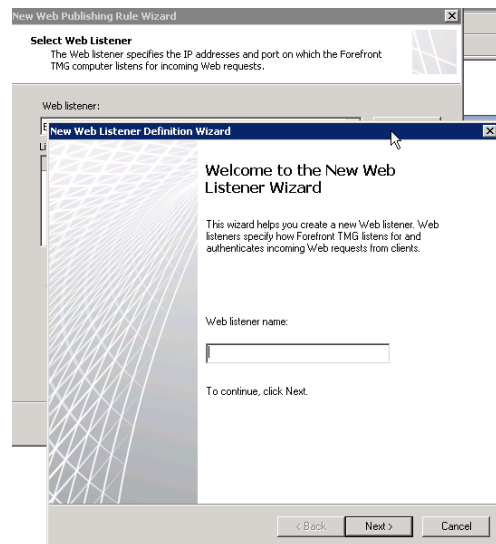


Otherwise, you will need to define a new web listener. When you click **New** above, the **Define**

Step 7 – Enable SNI

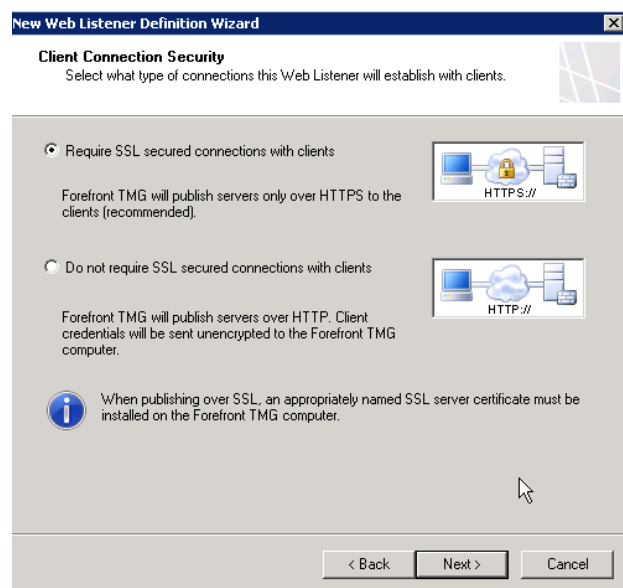


New Web Listener wizard is presented. Provide an appropriate name for the listener and click **Next**.



Step 8 – Client connection security

You must now define how the client will connect to TMG.



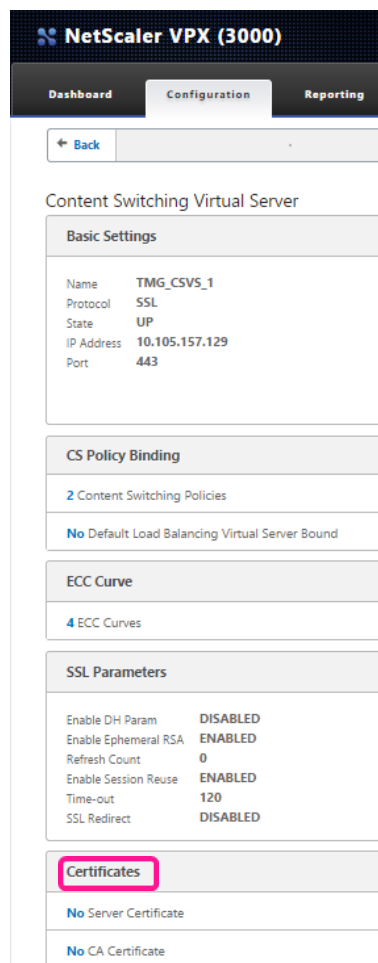
Here, you may choose between making SSL connections necessary and allowing non-SSL connections. Note that when publishing over SSL, an appropriate SSL server certificate must be installed on the TMG server.

After returning to the basic settings screen for the CS vserver, If SSL has been enabled, make sure the **SNI Enable** option in the **SSL Parameters** tab is selected. This will allow the CS vserver to serve the right certificate for each website.

With SNI enabled, NetScaler will parse certificates and extract the hostname from them, then pass the certificate corresponding to the hostname requested by the client. This eliminates the need to use wildcard or SAN certificates.

Step 8 – Adding certificates to the CS vserver

After Step 7, go back to the basic settings screen for your CS vserver. Open/move to the **Certificates** tab, then click on **Server Certificates**.



In the prompt that follows, add the server certificates for all the servers that you have defined as part of your multiple website setups. Make sure you select the **Server Certificate for SNI** option, so the server certificates can be served properly.

Step 9 - Define listener IP addresses

Name	Selected IPs
<input type="checkbox"/> External	<All IP addresses>
<input type="checkbox"/> Internal	<All IP addresses>
<input type="checkbox"/> Local Host	<All IP addresses>
<input type="checkbox"/> Quarantined VPN Clients	<All IP addresses>

The next screen asks you to define the IP addresses that the listener will listen for. Only requests from the networks specified here will be forwarded to the web servers defined earlier.

Step 10 – Link certificates for specified IPs

IP Address	Network	Server	Certificate
------------	---------	--------	-------------

Now you can link client certificates for SSL connections; you can choose to use a single certificate or assign certificates for individual IP addresses that will access the server.

Server Certificate Binding

Select Server Certificate*

Click to select > +

Server Certificate for SNI

Bind Close

This completes the configuration for multiple website publishing with NetScaler.

Further, access definitions (selective user access, selective network access, etc.) can be configured using specific policies on NetScaler. These are defined based upon the specific use case, and are not wizard based (however, the NetScaler policy definition GUI provides context-sensitive help).

NetScaler advantage

TMG only allows use of SAN or wildcard certificates, as it does not support SNI (subject name identification). With NetScaler, you can enable SNI or use wildcard or SAN certificates, in which case you do not need to enable SNI in Steps 7 and 8 and one common certificate may be used across websites. However, for greater security, it is recommended that SNI be used.

Step 11 – Client authentication method definition

The screenshot shows a window titled "New Web Listener Definition Wizard" with a close button (X) in the top right corner. The main heading is "Authentication Settings" with a sub-heading: "Select how clients will authenticate to Forefront TMG, and how Forefront TMG will validate their credentials." Below this, there are two sections. The first section is "Select how clients will provide credentials to Forefront TMG:" with a dropdown menu currently set to "No Authentication". The second section is "Select how Forefront TMG will validate client credentials:" with five radio button options: "Windows (Active Directory)" (selected), "LDAP (Active Directory)", "RADIUS", "RADIUS OTP", and "RSA SecurID". At the bottom of the dialog are three buttons: "< Back", "Next >", and "Cancel".

On the next screen, you can define how clients are authenticated with TMG. TMG supports LDAP/Active Directory, Radius or RSA SecurID-based authentication. **Note that NetScaler supports several other alternative authentication mechanisms as well, as indicated in our note earlier.**

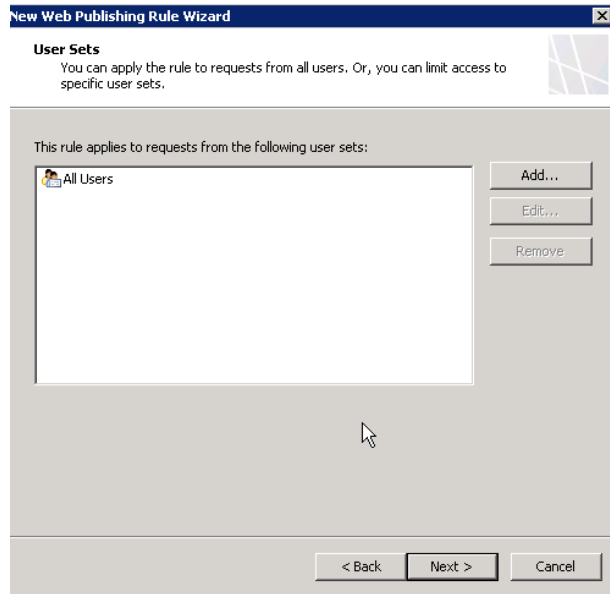
Since our deployment on TMG currently will not use any authentication scheme, we will leave it at **No Authentication**.

Step 12 - SSO

The screenshot shows a window titled "New Web Listener Definition Wizard" with a close button (X) in the top right corner. The main heading is "Single Sign On Settings" with a sub-heading: "Single Sign On (SSO) allows users to authenticate once to Forefront TMG to access all published Web sites that use this Web listener." Below this, there is a checkbox labeled "Enable SSO for Web sites published with this Web listener" which is currently unchecked. Below the checkbox is an information icon (i) with a message: "SSO is not available for the currently selected client authentication method. SSO is only available for HTML Form Authentication." At the bottom of the dialog are three buttons: "< Back", "Next >", and "Cancel".

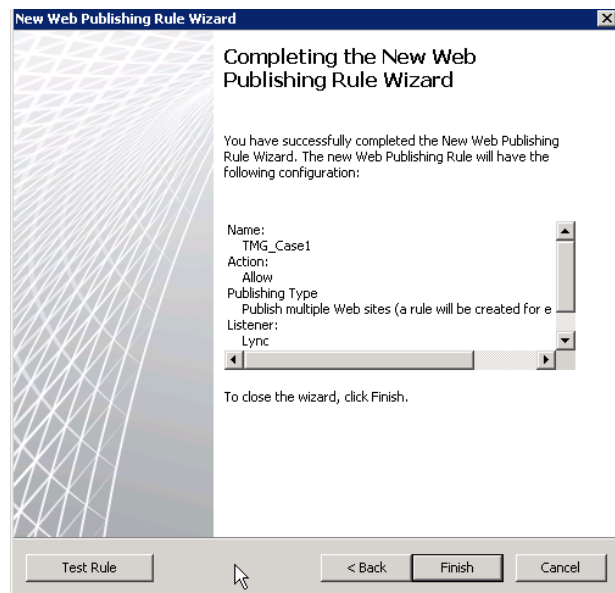
SSO is only available for HTML form-based authentication.

Step 14 – User set definition



In this step, you can define the users to whom this rule will apply. Essentially, these will be the only users who will have access to these websites.

Step 15 – Test rule and conclude the wizard



The final step allows you to test the rule you have created and concludes the wizard.

Additional information

SSL bridging and tunneling

SSL bridging and tunneling are two Forefront TMG features that are extremely important for most enterprises. In TMG parlance, SSL bridging refers to the termination or initiation of an SSL connection by TMG. As an example, when a client makes an HTTP request, TMG creates a request on the client's behalf and encrypts it. This request is then forwarded to the target web server over HTTPS. The web server responds appropriately and sends the requested object to TMG. It is then decrypted and sent to the client that requested the original HTTP object, thus making the process transparent to the user. Alternatively, TMG can decrypt HTTPS traffic originally sent by the client, inspect it and then re-encrypt it using a different certificate and have it sent on.

SSL tunneling, on the other hand, directs TMG to create a secure SSL tunnel from the internal client to the external server. Here, a typical connection is established as follows:

The client browser makes a request to the external website in the form: **https://<URL_Name>**

The request is sent to port 8080 on the Forefront TMG computer, as follows: **CONNECT <Server name>:443 HTTP/1.1.**

The Forefront TMG server connects to port 443 on the destination web server, and when the TCP connection is established, returns: **HTTP/1.1 200 connection established.**

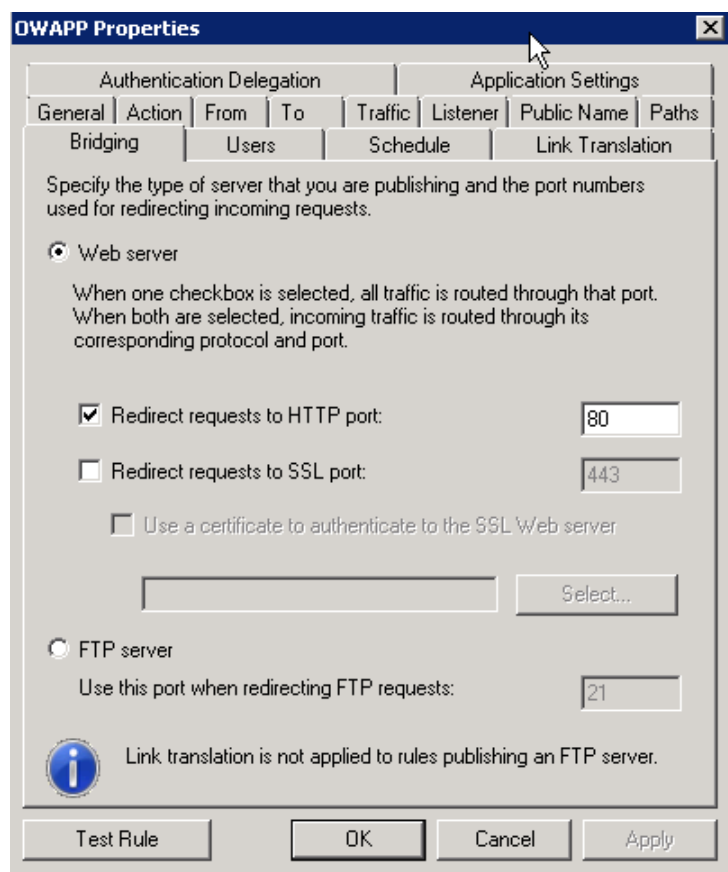
NetScaler allows this functionality to be duplicated; however, the terminology used is slightly different:

TMG feature	NetScaler equivalent
SSL bridging	SSL offloading
SSL tunneling	SSL bridging

Steps for deployment on TMG

SSL bridging

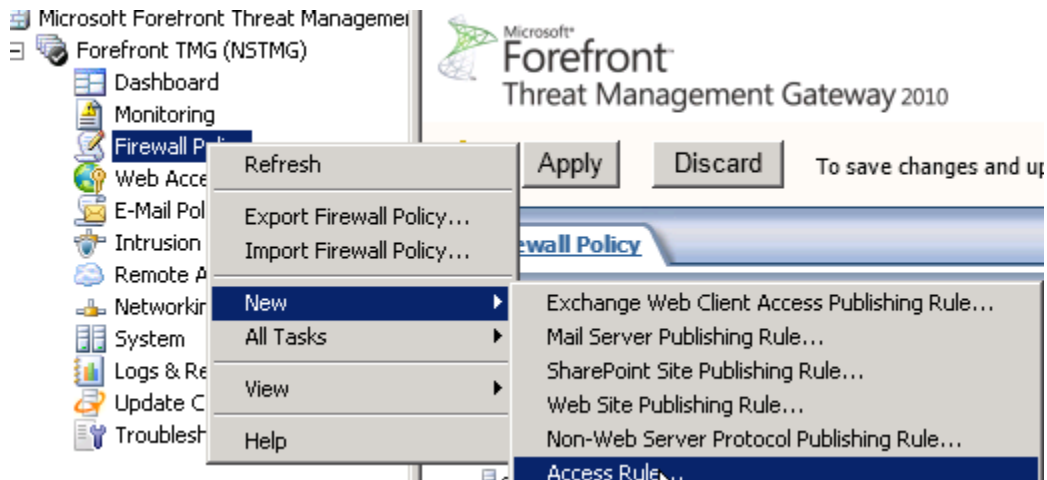
SSL bridging on TMG is linked to individual firewall policy rules. To enable bridging, choose the **Firewall Policy** tab in the panel on the left side of the screen, then select the appropriate firewall rule and right click on it. After selecting **Properties**, you will see the following screen:



Here, select the **Bridging** tab, then ensure that **Redirect requests to HTTP port** is not checked, that the **Redirect requests to SSL port check box** is selected and that 443 is the port, and then click **OK**.

SSL tunneling

To enable SSL tunneling, create a new access rule (or modify a pre-existing one) by right clicking on **Firewall Policy** in the left-hand pane on the TMG console; here, select **New > Access Rule**.



The **New Access Rule** wizard is initiated. Begin by providing a name for the new access rule. After clicking **Next**, choose **Allow** in the **Rule Action** window (this defines whether the rule is an allow or deny rule; as we are configuring SSL tunneling, this will be an allow rule). After clicking **Next**, in the **Protocols** window, choose **Selected Protocols** in the dropdown menu, **This rule applies to**; now, click **Add** and choose the **HTTPS Server** protocol under **Common Protocols**. Here, you can also choose different ports if the SSL traffic in your environment goes to a different port by using the **Ports** option in this window. In the next window, **Access Rule Sources**, add the networks that will provide the source traffic for this rule; this is the network containing the clients that will be accessing external resources using tunneling. To add a new network, click **Add** and choose the appropriate network from the options provided. On the next screen, **Access Rule Destinations**, select the network that contains the target sites that clients will access over this SSL tunnel by choosing the **Add** button, as with the last window prompt.

The next window is **User Sets**, where you can define the users to whom this rule will apply.

Note: You will observe after creating this rule that all the options in the **Malware Inspection** tab in the **Properties** window for this rule are disabled. This is because with SSL Tunneling, TMG cannot inspect the traffic as it is encrypted and TMG has no means of decrypting the same.

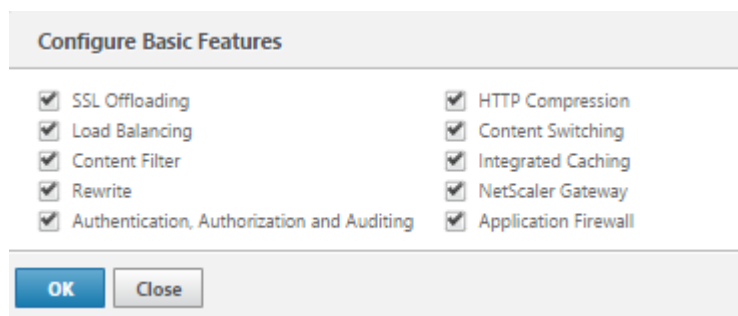
Steps for deployment on NetScaler

SSL offloading (equates to SSL bridging on TMG)

To configure SSL offloading, the following steps need to be completed:

1. You must enable SSL offloading on the NetScaler appliance.
2. Configure an SSL-based virtual server that will intercept SSL traffic, decrypt the traffic and forward it to a service that is bound to the virtual server.
3. To enable SSL offloading, you must import a valid certificate and key and bind the pair to the virtual server.

Step 1 - Enabling SSL offloading: Before beginning the configuration of SSL offloading on the NetScaler appliance, you must make sure the SSL offloading feature is enabled. To verify this, navigate to **System>Settings** in the left-hand navigation pane in the NetScaler GUI, then click **Configure Basic Features** under **Modes and Features**. In the next prompt, make sure **SSL Offloading** is checked. If not, enable it and in the **Enable/Disable Feature(s)?** message box, click **Yes**. A message appears in the status bar confirming that the feature has been enabled.



Step 2 - Configuring the SSL-based virtual server: To configure an SSL-based virtual server, follow these steps:

1. First add a service by navigating to **Traffic Management > Load Balancing > Services**
2. In the **Services** pane, do one of the following:
 - a. To create a new service, click **Add**.
 - b. To modify an existing SSL service, select the service, and then click **Open**.
3. In the **Basic Settings** dialog box, specify values for the following parameters (all four are required parameters):
 - a. Service Name – A uniquely identifiable name
 - b. IP Address – The IP address of the server running the service
 - c. Protocol – Define as SSL
 - d. Port – 443 (or an alternate port as per your enterprise configuration)

Click **OK**. In the **Services** pane, select the service that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Navigate to **Traffic Management > Load Balancing > Services**, create a service, and specify the protocol as SSL.

The CLI commands for achieving this are:

```
add service <name> (<IP> | <serverName>) <serviceType> <port>
show service <serviceName>
```

Parameter Descriptions (of commands listed in the CLI procedure)

name - Name for the service. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the service has been created.

IP - IP to assign to the service.

serverName - Name of the server that hosts the service.

serviceType - Protocol in which data is exchanged with the service.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, RPCSVR, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, SIP_UDP, DNS_TCP, ADNS_TCP, MYSQL, MSSQL, ORACLE, RADIUS, RDP, DIAMETER, SSL_DIAMETER, TFTP (For the SSL Offload feature, use SSL, for SSL Bridging use SSL_BRIDGE)

Port - Port number of the service.

After configuring the service, configure the SSL-based virtual server by navigating to **Traffic Management > Load Balancing > Virtual Servers**, creating a new virtual server by clicking **Add**. Here, in the **Basic Settings** screen that comes up, provide a name, define the protocol as SSL and provide an IP address. Since the protocol is SSL, the default port chosen is 443, although you may pick an alternate port as per your configuration.

Load Balancing Virtual Server

The screenshot shows a 'Basic Settings' dialog box for configuring a virtual server. It includes the following fields and controls:

- Name***: A text input field with a question mark icon to its right.
- Protocol***: A dropdown menu currently set to 'SSL'.
- IP Address Type***: A dropdown menu currently set to 'IP Address'.
- IP Address***: A text input field with a dotted placeholder (e.g., '. . .') and an 'IPv6' checkbox to its right.
- Port***: A text input field containing the value '443'.
- At the bottom, there is a 'More' link with a right-pointing arrow, and two buttons: 'OK' and 'Cancel'.

Next, you will need to bind the service created earlier to this SSL-based virtual server. To do this, navigate to **Traffic Management > Load Balancing > Virtual Servers**. Here, choose the virtual server created earlier by its name and in the **Basic settings** screen that follows, click in the **Services and Service Groups** section on **No Load Balancing Virtual Server Service Binding** section to bind the service you created earlier to the virtual server.

Basic Settings	
Name	TMG_formbased_sp
Protocol	HTTP
State	Down
Traffic Domain	0
Listen Priority	-
Listen Policy Expression	-
Range	1
Redirection Mode	IP
RHI State	PASSIVE
AppFlow Logging	ENABLED

Services and Service Groups	
No Load Balancing Virtual Server Service Binding	>
No Load Balancing Virtual Server ServiceGroup Binding	>

Next, you need to configure a certificate-key pair that this virtual server can use for SSL offloading. To do this, navigate to **Traffic Management > SSL > Certificates** and click on **Install** to configure a certificate-key pair. Clicking on Install will present the screen shown below. Enter the required settings as per your configuration.

Install Certificate

Certificate-Key Pair Name*

Certificate and Key files are stored in the folder /nsconfig/ssl/ on appliance.

Certificate File Name*

 Browse +

Key File Name

 Browse +

Certificate Format

PEM DER

Password

Certificate Bundle

Notify When Expires

Notification Period

Install Close

Next, you should bind this certificate-key pair to the virtual server created earlier. To do so, navigate to **Traffic Management > Load Balancing > Virtual Servers**.

Open the SSL virtual server by selecting it and clicking **Edit** and, in Advanced Settings, click **Certificates**, then **Server Certificate**. In the screen that follows, click **Add Binding**. Here, you can bind a server certificate to the virtual server. To add a server certificate as an SNI certificate, select **Server Certificate for SNI**.

This completes the configuration for SSL offloading on the NetScaler appliance.

SSL bridging (equates to SSL tunneling on TMG)

An SSL bridge configured on the NetScaler appliance enables it to bridge all secure traffic between the SSL client and the SSL server. The appliance does not offload or accelerate the bridged traffic, nor does it perform encryption or decryption. Only load balancing is done by the appliance. The SSL server must handle all SSL-related

processing. Features such as content switching, SureConnect and cache redirection do not work, because the traffic passing through the appliance is encrypted.

Because the appliance does not carry out any SSL processing in an SSL bridging setup, there is no need for SSL certificates. Before you configure SSL bridging, first enable SSL and load balancing on the appliance (per the section above). Then, create **SSL_Bridge** services and bind them to an **SSL_Bridge** virtual server. For this, follow the same steps for service and virtual server creation as described for SSL offloading above, only instead of selecting SSL as service type during service configuration, select **SSL_BRIDGE**.

Configure the load balancing feature to maintain server persistency for secure requests. To achieve this, set the persistence setting to **SSLSESSION** in the **Basic settings** for the virtual server that you have created.

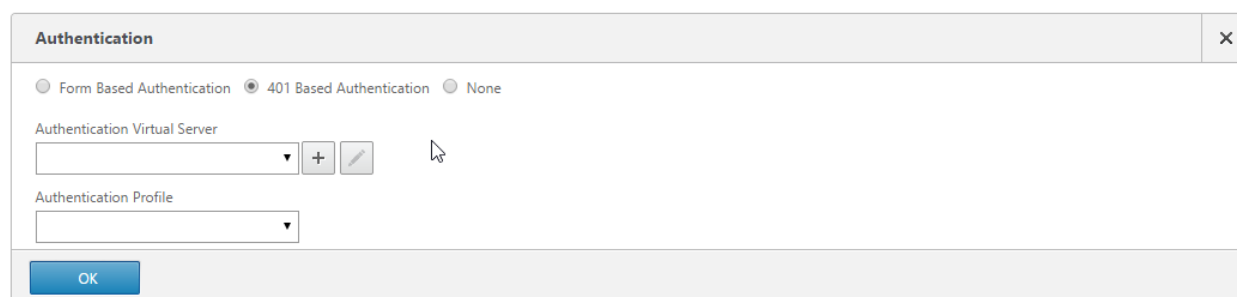
This completes the configuration for SSL bridging/tunneling on the NetScaler appliance.

Authentication

Steps 11 and 12 for the TMG configuration process look at authentication definition. Although these steps are not essential for our current use case, they will become relevant and be expanded upon in a subsequent guide.

The following section gives a basic overview for enabling authentication on NetScaler for website publishing. We will provide a detailed description of the various authentication options available in a subsequent guide.

NetScaler supports multiple authentication formats for user authentication with websites. For enabling authentication, first expand the **Authentication** tab from the side panel on the **Basic settings** screen on the LB vserver where the website that requires authentication is linked.



The screenshot shows the 'Authentication' configuration dialog box. At the top, there are three radio buttons: 'Form Based Authentication', '401 Based Authentication' (which is selected), and 'None'. Below this, there is a section for 'Authentication Virtual Server' with a dropdown menu, a '+' button, and a pencil icon. Below that is a section for 'Authentication Profile' with a dropdown menu. At the bottom left, there is a blue 'OK' button.

Select the appropriate authentication mechanism (Form Based or 401(basic or NTLM) authentication), then define the authentication virtual server and profile.

Defining the Authentication vserver and profile

To define the authentication settings for the LB vserver, navigate to **Security>AAA Application Traffic**.

- + System
- + AppExpert
- + Traffic Management
- + Optimization
- Security
 - AAA - Application Traffic
 - Virtual Servers**
 - Authentication Profile
 - Groups
 - Users
 - KCD Accounts
 - + Policies
 - + Application Firewall
 - + Protection Features

NetScaler > Security > AAA - Application Traffic > Authentication Virtual Servers

Add
Edit
Delete
Statistics
Action ▾

Name	State
Exchange_2013_401_Auth_Vserver	● Up
ActiveSync-AAA	● Up
Negotiate_Authentication	● Down

Here, you must define the authentication virtual server as well as the profile and relevant policies.

It is helpful to first define the authentication policies, however, as they are bound to the virtual server. The authentication policies can be defined at **Security>AAA – Application Traffic>Policies>Authentication>Basic Policies**.

NetScaler advantage

As compared to TMG, NetScaler supports several additional authentication profiles. In addition to Basic/Local, NTLM/AD and RADIUS that are supported by TMG, NetScaler supports CERT (certificate based), TACACS, web and SAML/SAML IDP-based authentication, thus providing significantly improved functionality.

Authentication Virtual Server

Basic Settings				✎
Name	IP Address	Port	Authentication Domain	
Exchange_2013_401_Auth_Vserver	10.105.157.94	443	ctxns.net	
Certificates				
1 Server Certificate				>
No CA Certificate				>
Advanced Authentication Policies				
No Authentication Policy				>
Basic Authentication Policies				+
Primary Authentication				
1 LDAP Policy				>
Secondary Authentication				
1 RADIUS Policy				>
401 Based Virtual Servers				×
No Load Balancing Virtual Server				>
No Content Switching Virtual Server				>
Form Based Virtual Servers				×
1 Load Balancing Virtual Server				>
No Content Switching Virtual Server				>

Each authentication profile will require settings appropriate for that authentication type. After creation of the requisite policies, they can be added to the authentication virtual server, along with information on the LB or CS virtual servers that are linked to this authentication virtual server.

Once definition of the authentication virtual server is complete, it can be bound to the website LB vserver. Authentication will be described in greater detail in a subsequent guide.

Conclusion

Citrix NetScaler provides a complete replacement for Microsoft Forefront TMG for organizations looking to securely host multiple, load balanced websites. NetScaler, as indicated through various callouts in this guide, presents several benefits over TMG. Further, the deployment of similar functionality on NetScaler is not only more flexible in scope than on TMG, but also much simpler.

About Citrix

Citrix (NASDAQ:CTXS) is leading the transition to software-defining the workplace, uniting virtualization, mobility management, networking and SaaS solutions to enable new ways for businesses and people to work better. Citrix solutions power business mobility through secure, mobile workspaces that provide people with instant access to apps, desktops, data and communications on any device, over any network and cloud. With annual revenue in 2014 of \$3.14 billion, Citrix solutions are in use at more than 330,000 organizations and by over 100 million users globally. Learn more at www.citrix.com

Copyright © 2015 Citrix Systems, Inc. All rights reserved. Citrix, NetScaler and NetScaler VPX are trademarks of Citrix Systems, Inc. and/or one of its subsidiaries, and may be registered in the U.S. and other countries. Other product and company names mentioned herein may be trademarks of their respective companies.

Corporate Headquarters
Fort Lauderdale, FL, USA

Silicon Valley Headquarters
Santa Clara, CA, USA

EMEA Headquarters
Schaffhausen, Switzerland

India Development Center
Bangalore, India

Online Division Headquarters
Santa Barbara, CA, USA

Pacific Headquarters
Hong Kong, China

Latin America Headquarters
Coral Gables, FL, USA

UK Development Center
Chalfont, United Kingdom

