

NetScaler Gateway

MARIUS SANDBU

Revision:

Version 1.03 (07.04/2016)

About the author:

Marius Sandbu works as a Cloud Architect for Exclusive Networks in Norway, where he focuses on software-defined datacenter, end-user computing and cloud technologies. He is a Microsoft Azure MVP, Veeam Vanguard and Vmware vExpert and the author of, Implementing NetScaler VPX and Mastering NetScaler VPX.

He can be contacted on twitter @msandbu or on his email msandbu@bigtec.com

Marius's blogs at <http://msandbu.wordpress.com>

Information about this eBook

This short eBook is to cover the most the different configuration options that are possible with NetScaler Gateway and also dig into Unified Gateway which is part of NetScaler version 11. This is intended for consultants or work with NetScaler Gateway and want to use this as a reference guide for troubleshooting or checking configuration.

The book is split into different sections, for instance there are separate section for ICA-proxy setup and another for Clientless Access and Full VPN. Some sections are also just grouped together because of my inability to group properly.

This book is not by any means a full guide to NetScaler Gateway, but I am dependent on feedback to make it even better. If you have any feedback, please send it to

Special thanks to my reviewers!

- Daniel Wedel
- Carl Stalhood
- Carl Behrent
- Dave Brett

Any feedback can be directed to my email msandbu@gmail.com

Note: that the information presented in this eBook is based on NetScaler version 11.0, XenDesktop 7.8 and Storefront 3.5, unless stated otherwise.

Content

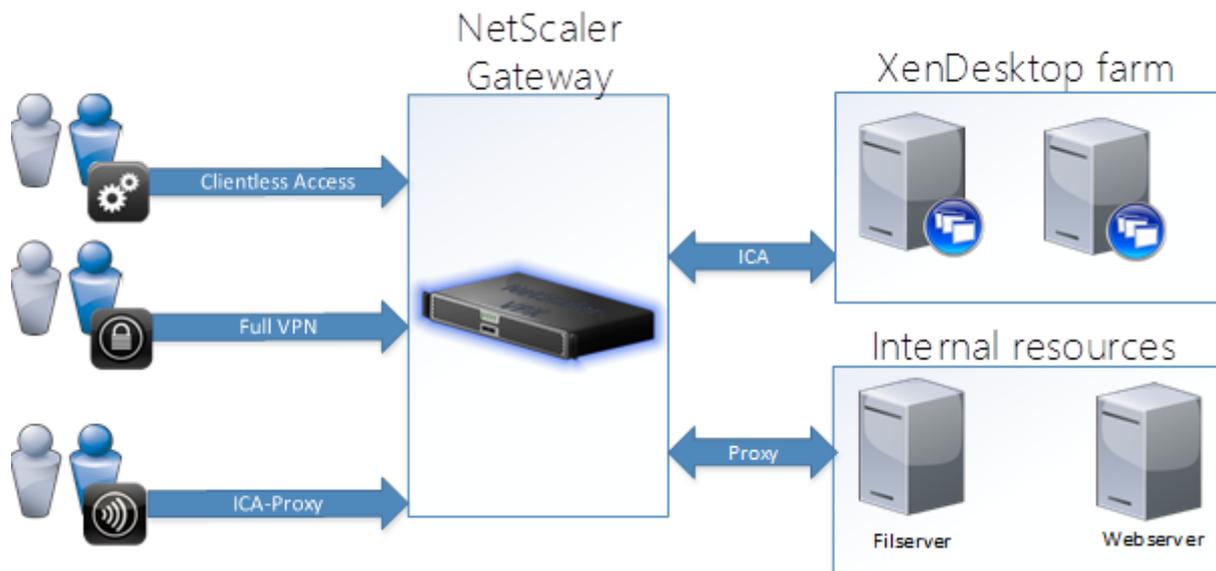
NetScaler Gateway basics	6
Licensing and editions	8
When to use what?	11
NetScaler and traffic flow	12
General settings for NetScaler	14
External authentication for administrators.....	14
Setting up ICA-proxy.....	16
ICA-proxy traffic Flow	16
Virtual Server setup	18
Certificates	20
Authentication.....	22
SSL Settings	25
Profiles.....	26
TCP Profiles	27
Published Applications	32
Policies	32
Citrix Receiver policy.....	33
Citrix Receiver for Web Policy.....	34
Storefront	35
Summary ICA-proxy.....	38
ICA Proxy with two armed.....	38
Double-hop configuration	41
Framehawk and Audio over DTLS.....	43
RDP Proxy	45
GSLB and Zone feature.....	50
GSLB Basics.....	50
Authoritative DNS	52
Zone based GSLB deployment	53
VPN and Endpoint analysis.....	53
Full VPN with endpoint scanning	54
Preauthentication policy	54
Session policy	60

Split tunneling.....	61
Client IP pools.....	62
Clientless Access	62
Adding resources.....	65
Binding the features together	67
Unified Gateway.....	68
Smart Access – Access Policy.....	73
Smart Control – ICA Policies	75
Group Based Access	77
High availability	79
Cross-subnet High-availability	83
Failsafe mode	83
VMAC	84
Upgrading.....	86
Portal customization.....	88
Binding an EULA to the Portal.....	90
Security settings	90
Authentication and Authorization.....	99
SAML Authentication	99
Allow password change from NetScaler Gateway.....	104
Allow password change from Storefront	106
Multifactor authentication	107
Authorization.....	112
Troubleshooting.....	115
Endpoint Access.....	115
Name resolution not working	115
ICA-proxy	116
Cannot complete your request	116
Your logon has expired	117
Unknown Client error 1110.....	117
Cannot Start Desktop "COMPUTERNAME"	118
Error: Login exceeds maximum allowed users.....	118
Http/1.1 Internal Server Error 43531.....	119

403 - Forbidden: Access is denied	119
Authentication.....	119
Other design examples.....	121
Multitenant ICA-Proxy	121
Monitoring.....	125
Insight Center.....	125
Command Center	127
Goliath IT analytics for NetScaler.....	128
System Center Operations Manager	130

NetScaler Gateway basics

NetScaler Gateway is a feature, which delivers remote access for end users. It can either be in form of remote access using Citrix Receiver, where we have the NetScaler gateway to proxy connections to backend XenDesktop servers. It can also be in form of clientless access meaning that we can use a regular web browser to get access to for instance internal web resources or even files. We can also use it for full VPN access meaning that our endpoint becomes part of an internal network and allows access to communicate directly with internal resources over a secure VPN tunnel.

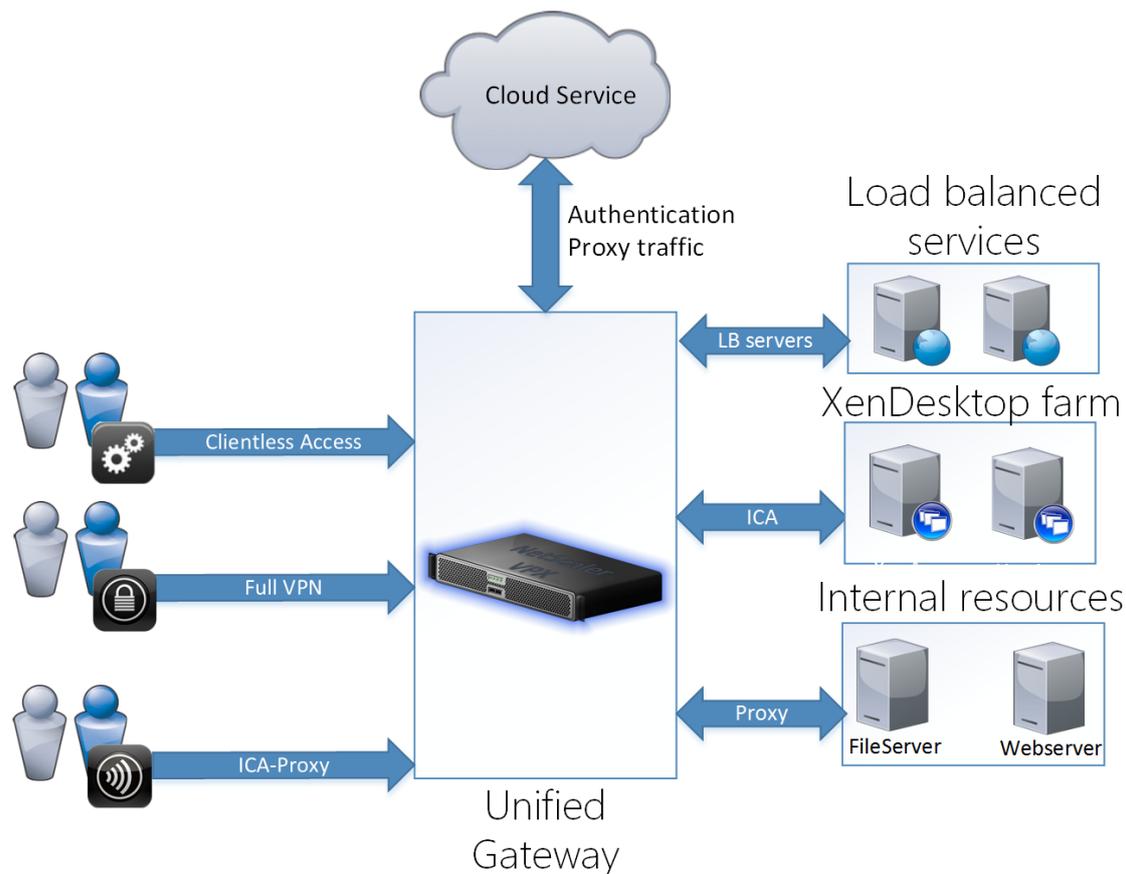


NOTE: NetScaler Gateway is one of the more common used features within Citrix NetScaler. Either it can be used as a feature on the NetScaler VPX/MPX or we can buy the NetScaler Gateway VPX/MPX, which only licensed to do NetScaler Gateway.

So for instance if we are using Citrix Receiver for remote access, it will connect directly to the Gateway virtual server which will then establish a connection with the backend XenDesktop farm. If we use the full VPN client, either we can be using the NetScaler as a source IP to browse internal resources, or we can be given an IP from a DHCP scope. We can also use the clientless access, which gives us SSL VPN over a regular Internet Browser and allows us to browse internal web resources and file servers.

In NetScaler 11, Citrix introduced something called Unified Gateway, which allowed us to aggregate load balanced web services, cloud services and internal Citrix applications in a unified app portal.

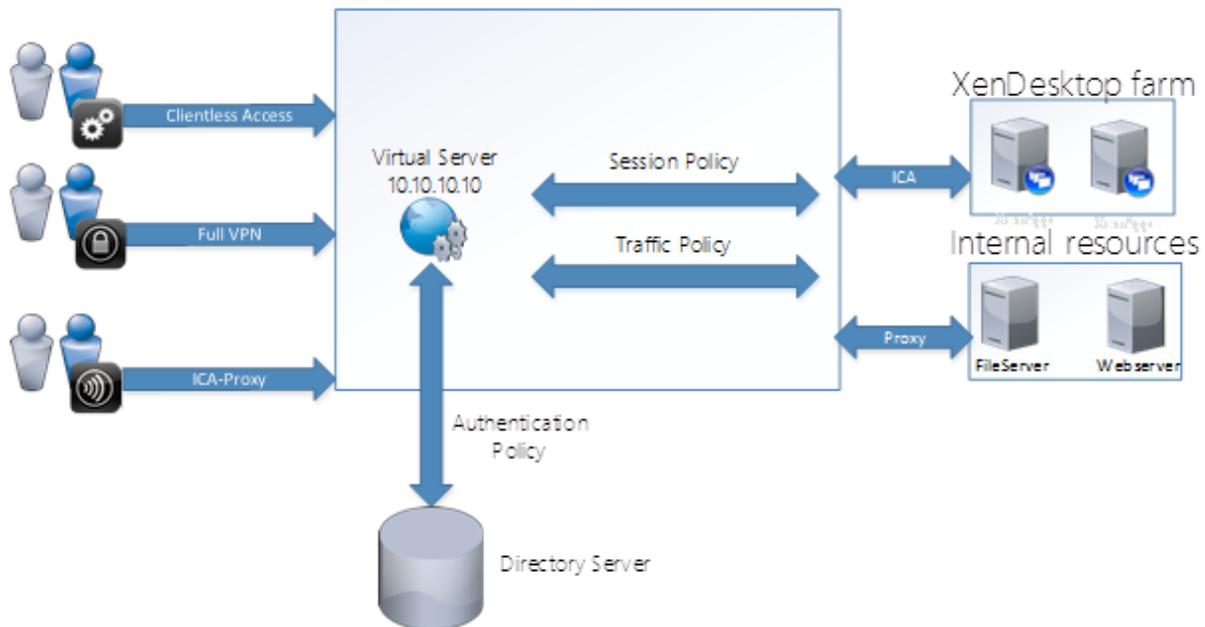
Unified Gateway leverages two additional features, content switching and AAA. The AAA module is used to deliver SSO against different resource such as internal load balanced resource and to cloud applications like Salesforce or Office365. Therefore, it is important to remember that Unified Gateway is not a feature available in NetScaler Gateway MPX/VPX



There will be more about Unified Gateway later in the eBook.

NetScaler Gateway is essentially a virtual server, which listens to requests on port 443 by default, and depending on the configuration can act as an ICA Proxy only virtual server or as multiple purpose remote access solution. When a user tries to connect to the virtual server, they will be asked to authenticate against the authentication policy which triggered, after successful authentication, the user will be processed against different policies and which case might allow them to setup an ICA proxy session with a backend XenDesktop server or full VPN access.

All of the ICA proxy and most of the VPN setup and configuration is mostly done using Session policies, where we define the address of Storefront and how the client should behave. Here we can also specify some of the particular VPN settings as well, some of the VPN settings are also done at the virtual server level, for instance if a virtual server should only be configured as an ICA proxy server or if it can be used for more of the advanced features like SSL VPN and or full VPN. We also have traffic policies, which are using to define for instance SSO properties to backend resources, enforce network traffic rules and disabling certain features as such. Therefore, this is pretty much the essence in how NetScaler Gateway looks like and behaves.



Licensing and editions

NetScaler Gateway can be used as a feature on a regular NetScaler appliance (running either Standard, Enterprise or Datacenter edition) or it can be used as a separate appliance either NetScaler Gateway MPX which is a physical appliance or NetScaler Gateway VPX which is a virtual appliance. The difference between the NetScaler Gateway appliance and the regular NetScaler is that the Gateway appliance ONLY has the Gateway feature.

Now we have two different licenses for use with NetScaler gateway, first thing we need is the platform license to be able to use the NetScaler platform and activate the gateway feature and the other is called Universal licenses, which enables additional features. Important thing to remember is that the universal license is optional depending if we need the features, the platform license is mandatory.

The regular NetScaler appliance physical or virtual platform is licensed using hostID, and the Gateway feature is included as a sub feature. The hostID of the appliance can be retrieved from the CLI using the **show hardware** command, which then needs to be entered using the Citrix licensing portal. If we use a NetScaler Gateway appliance, it needs to be licensed using hostname, which can be configured and retrieved from the CLI, using the command **set** and **show hostname**.

Both these options will give us a platform license. Now if we just use the platform license, we get the following features:

- ICA Proxy
- NetScaler (High-availability)
- Central administration using Command Center
- Unlimited virtual servers

Note that there is no user limit with the platform license, meaning that if we allocate a platform license to a NetScaler it is bound to the appliance. Which also means that we have no licensing user limit to the ICA proxy solution, it is only based upon the amount of users the NetScaler can handle.

Universal license

By default, all NetScaler appliances (NetScaler Gateway/NetScaler Standard/NetScaler Enterprise) comes with five Universal licenses. NetScaler platinum comes with 100 Universal licenses. If they want more users, they need to buy additional Universal licenses, which comes at a concurrent user license.

A Universal license is required if we want to use a NetScaler Gateway with the following features

- SSL VPN
- Full VPN Access
- MicroVPN for XenMobile
- Cloudbridge integration
- Endpoint analysis
- SmartAccess
- Secure Access to ShareFile / XenMobile

Universal licenses are also licensed using hostname when defining this in the Citrix licensing portal. Licenses can be simply added using the GUI by going into the management portal, System → Licenses → Manage licenses → Add New License

After a license has been added we can see which features we have access to (depending on the platform license) and the Maximum amount of NetScaler Gateway Users Allowed, which specifies the amount of concurrent universal licenses we have.

NetScaler > System > Licenses			
Manage Licenses...			
License type	Platinum	Model ID	1000
Load Balancing	✓	SSL Offloading	✓
Content Switching	✓	Cache Redirection	✓
Global Server Load Balancing	✓	GSLB Proximity	✓
Authentication, Authorization and Auditing	✓	NetScaler Gateway	✓
Maximum NetScaler Gateway Users Allowed	5	Maximum ICA Users Allowed	Unlimited

How a NetScaler chooses from the different license is defined at the virtual server level. A virtual server can be either in Basic mode or Smart Access mode. If a virtual server is in basic mode, it uses the platform licenses and we are given access to the ICA proxy feature. In version 11 this is defined as ICA only mode which can be enabled/disabled under the virtual server configuration

NetScaler Gateway → Virtual Servers → Edit → Basic Settings

VPN Virtual Server

Basic Settings
Name

IP Address Type
IP Address ▼
IPAddress*
 IPv6
Port
443
RDP Server Profile

Maximum Users
0
Max Login Attempts

Failed Login Timeout

 ICA Only
 Enable Authentication

If we have this enabled, we will not be able to use features, which depend on Universal licenses like, SSL VPN or Full VPN features. If we remove this checkbox, it will be enabled as a Smart Access virtual server and will start using universal licenses when a user connects.

Another important thing to consider when setting up NetScaler Gateway is the amount of supported users per appliance.

For Gateway deployments on a virtual appliance, meaning the NetScaler Gateway VPX or the NetScaler VPX Citrix supports up to 1500 concurrent VPN or ICA users, while on the physical appliances Citrix supports from 5000 to 35,000 concurrent users on VPN or ICA. Now the supported amount of users on virtual or physical appliances all depend on the network, license, bandwidth usage and so on.

Important to remember that this restriction is based upon access to SSL chips, which a regular VPX does not have. It is also important to remember that the limit of 1500 is based upon the resources available on the underlying hypervisor and packet engine CPUs

NOTE: You can read more about packet engines and SSL performance on the VPX here
→ <http://bit.ly/24RDmv1>

When to use what?

Just to use some examples on when to use what in a Gateway scenario and what kind of license we need.

1: Just need Citrix Receiver remote access for our end-users, we are about 500 users.

A NetScaler Gateway VPX would suffice or a Gateway MPX if the customer wants physical hardware instead of placing the load on the virtualization platform. Important to remember that the restrictions on the VPX in terms of SSL performance.

2: Need remote access for our users, but will be a mix of Citrix Access and VPN for 5000 users.

We can setup two virtual servers, where one is in smart access mode enabled for VPN, which then will use Universal licenses. We also setup another virtual server, which is used for ICA-proxy, will be setup in basic mode, and will not require any universal licenses. This way we can save money if there is only a few of our users which are going to use VPN. On the other hand, this will give us two different IP addresses and FQDN, which users have to remember. If all our users are going to be using VPN, then the best practice is to setup one virtual server in smart access mode but then it is important that we need to have sufficient universal licenses for all our users.

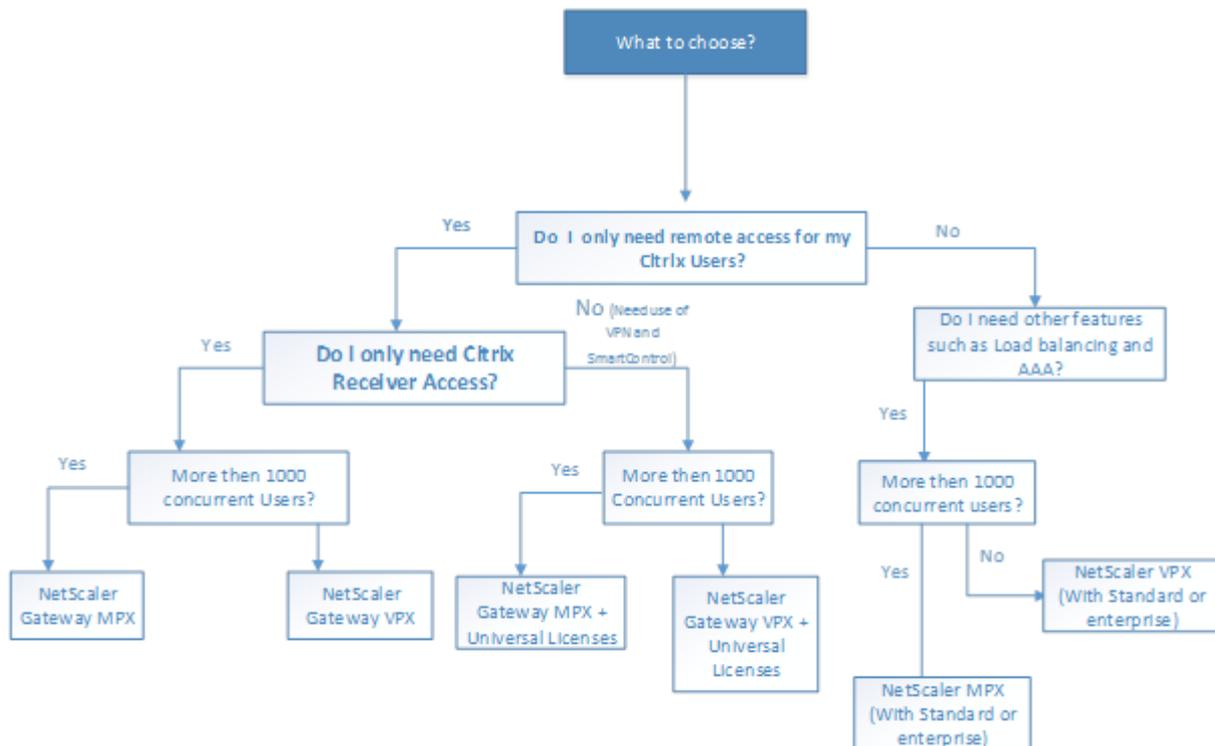
3: Have an existing NetScaler used for load balancing, need to setup a remote access for Citrix

In this case, we already have a NetScaler license in place, just need to create a virtual server in ICA-only mode and define our policies.

4: We have an existing NetScaler Gateway VPX used for ICA-only but want to use Unified Gateway to access SaaS based applications

First of we need to upgrade our license to a NetScaler Enterprise since Unified Gateway uses the AAA module which is only part of Enterprise, then we need to recreate a Virtual Server which is using Unified Gateway. Unified Gateway also uses Universal Licenses so therefore we need to buy licenses that are more concurrent as well.

Below is an example decision tree, which shows when to choose what based upon the requirements. Note that not all examples are included in this tree but will give you some indication.



NetScaler and traffic flow

In order to properly configure a NetScaler, it is important to understand the traffic flow it operates in. By default, IP-addresses are not bound to any particular interface, if there is a request for some content from a IP-address that is owned by the NetScaler it will respond on that interface where the request comes from.

NetScaler operates at Layer 3 and operates with many different IP-addresses. There are 3 primary addresses

- NSIP
- VIP
- SNIP

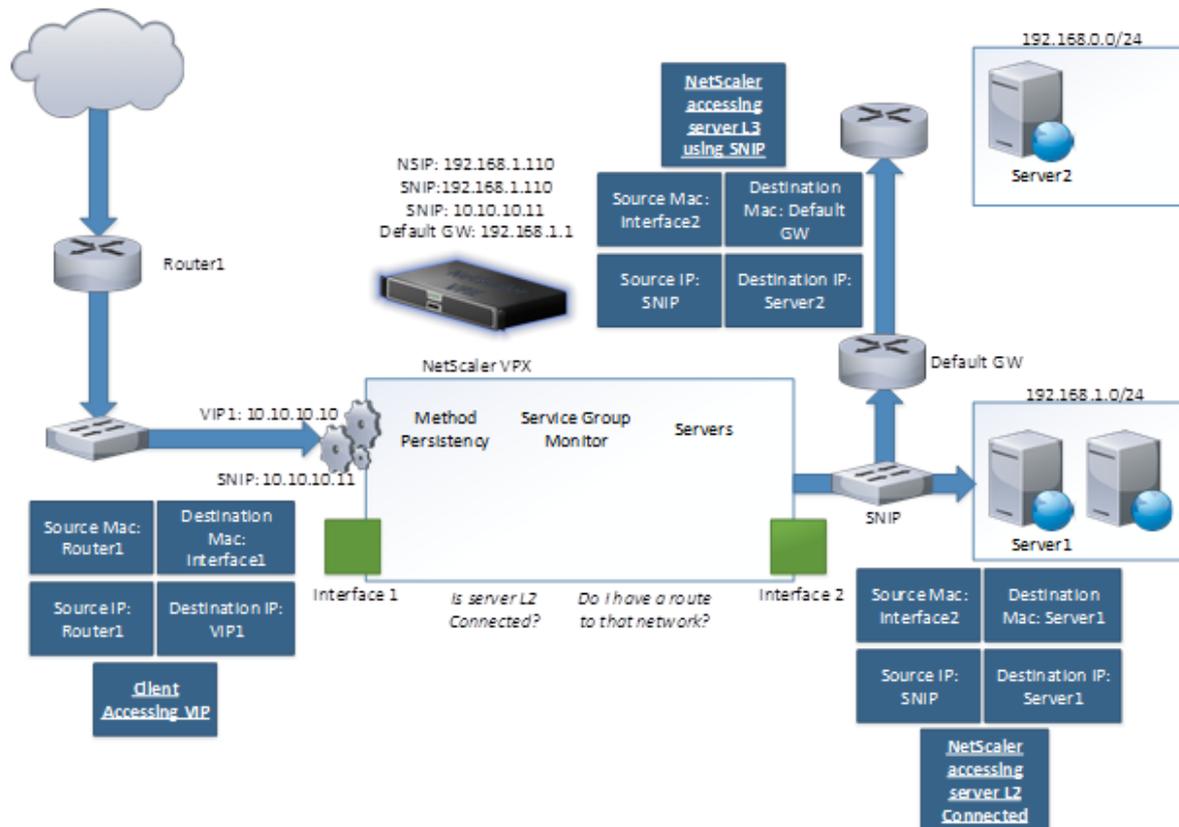
NSIP is used for management, authentication traffic, some monitoring traffic, DNS and Syslog. We can have only one NSIP and if we need to change this IP-address we need to restart the appliance

VIP is a packet processing IP, it represents a virtual service like a load balanced service or a NetScaler Gateway. It only handles incoming traffic and applies logic and forwards it to a backend IP. There is only one VIP per virtual server, but we can have as many of these as we want.

SNIP is a packet generating IP, it is typically setup in order for the NetScaler to communicate with backend resource on layer 2. For instance, if we want to load-balance two IIS web servers located on 10.10.10.10/24 network we would typically have an SNIP located on the 10.10.10.10/24 network which is used for communication between the

NetScaler and the web-servers. SNIP can also be used to traverse different subnets as long as there is a route present. Remember also when we create a SNIP on a NetScaler it will automatically create a DIRECT ROUTE to that particular layer 2 network in its routing tables.

So just to give an example of how traffic will flow in a NetScaler setup.



We have a virtual server (Load balanced server) which is represented by the IP 10.10.10.10. It has defined a load balancing method, persistency and attached a number of servers in a service group which as a monitor attached to it, to make sure that the backend servers are up and running. The monitor is being processed by the SNIP 192.168.1.110 to communicate with the backend servers on the different subnets.

1. Traffic hits the 10.10.10.10 IP from endpoint 1.1.1.1
2. VIP processes the packet with the logic it has attached forwards it internally
3. Locates the closest SNIP to the backend resources on 192.168.1.0/24
4. SNIP initiates a connection to the backend server where the source IP will be the SNIP.
5. Backend server responds back to the SNIP
6. SNIP forwards the packet back based upon the internal session table
7. VIP responds back to the endpoint on 1.1.1.1 with the content.

Now there are some things that are important to remember, a VIP is only a packet processing IP there it cannot on itself send data back on the VIP. In this case we need to

have a SNIP present on the same subnet to get the packet generating capabilities on that subnet.

Note that the SNIP on the public facing interface will never be the source of the packets going back, but it is just needed to be able to process packets on the way back. Now as I also mentioned that a SNIP can be on the same subnet which the backend resource resides or in another subnet as long as it has a route present. In some case you would need to deploy a SNIP in a DMZ and setup static routes to the different subnet so the traffic would be processed by the company firewall as well. We will cover some different network designs when we come into the deployment of the gateways.

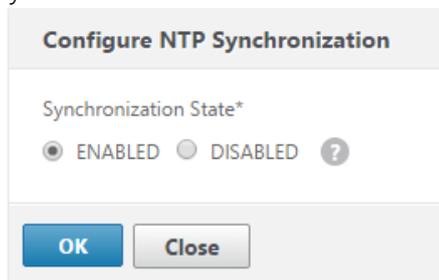
General settings for NetScaler

Before we go ahead and start configuring NetScaler Gateway, there are some basic settings we should get in place to ensure that is going to work properly.

- Add DNS servers (This can be done under Traffic Management → Name Servers → Click Add and enter an IP-address of the server. If we go back to the Name Servers menu after adding a DNS server, it shows as up. This uses simple ICMP to check if the DNS is up or not and all traffic going to the DNS server is sourced from the NSIP)

NOTE: We can setup a load balanced DNS server on the NetScaler, this will then in turn set the source IP from SNIP closest to the DNS server

- Change the time zone and setup NTP sync (Time zone can be changed under System → Settings → Change time zone and click OK (Do not restart right away) then go into NTP servers → Click Add, enter FQDN name of the server and click OK. After adding a server mark the server, select Action and choose Configure NTP Synchronization and mark it is enabled



External authentication for administrators

Setting up External Authentication allows administrator to authenticate to NetScaler using for instance their Active Directory users. In order to set this up go into the GUI → System → Authentication → LDAP → Servers and then click Add.

From there define an Active Directory server

The screenshot shows a configuration form for an Active Directory server. On the left, there are radio buttons for 'Server Name' and 'Server IP', with 'Server IP' selected. Below this is a text field for 'IP Address*' containing '10 . 217 . 215 . 101' and a checkbox for 'IPv6'. Underneath is a dropdown for 'Security Type*' set to 'PLAINTEXT', and a text field for 'Port*' set to '389'. On the right, there is a dropdown for 'Server Type*' set to 'AD', a text field for 'Time-out (seconds)' set to '3', and a checked checkbox for 'Authentication'.

Then we need to define an active directory base DN where our administrator users are located which will be the scope when doing LDAP bind. We also need to define an Active Directory service account which is being used to query AD.

The screenshot shows a configuration form for LDAP bind. On the left, there is a text field for 'Base DN (location of users)' containing 'CN=users,DC=test,DC=local', and a text field for 'Administrator Bind DN' containing 'ns_srv'. On the right, there is a checked checkbox for 'BindDN Password', followed by two password fields for 'Administrator Password' and 'Confirm Administrator Password', both containing masked characters. At the bottom right, there is a blue link labeled 'Retrieve Attributes'.

And finally we need to define a logon name attribute and group attributes so we can get information about the different groups and which ones are allowed access and not.

The screenshot shows a configuration form for LDAP search and authentication. On the left, there is a dropdown for 'Server Logon Name Attribute' set to 'sAMAccountName', a text field for 'Search Filter', a dropdown for 'Group Attribute' set to 'memberOf', and a dropdown for 'Sub Attribute Name' set to 'cn'. On the right, there is a text field for 'Default Authentication Group', a checked checkbox for 'User Required', an unchecked checkbox for 'Referrals', a text field for 'Maximum Referral Level' set to '1', a dropdown for 'Referral DNS Lookup' set to 'A-REC', and an unchecked checkbox for 'Validate LDAP Server Certificate'.

After we are done adding the information click Create. Now go back to the **GUI → System → Authentication → LDAP →** and then click on Policies and click Add.

Name the new policy, specify the LDAP server we just created and use the expression "ns_true"

Name*

Server*
 +

Expression*

Then click Create. Now go back to GUI → System → Authentication → LDAP → Policies → Global Bindings, from there click Add bindings. Then choose the policy we just created and make just use the default priority. The policy should appear like so

System Global Authentication LDAP Policy Binding

Priority	Server	Expression
100	AD	ns_true

Setting up ICA-proxy

This Section will focus on all of the aspects of setting up Citrix Remote Access using ICA-Proxy. This feature does not require any additional licenses besides the platform license for NetScaler Gateway or using it for instance as a sub feature of NetScaler. This section does a step-by-step approach and focuses on setting it up and configuring some of the other pieces such as TCP profiles to ensure optimal traffic flow. It also does a bit more in-depth on TCP profiles to allow in-depth knowledge about the TCP protocol and traffic flow.

Now before we go into the configuring part we should understand how the traffic flow of ICA-proxy is going to work after everything is properly configured from an end client perspective.

NOTE: This scenario was created against a XenDesktop 7.8 environment, using NetScaler version 11.64 and Storefront 3.5. Also important to note that this only requires a standard platform license, no additional licenses required.

ICA-proxy traffic Flow

This is a simple chart showing how traffic is going to originate from the endclient and eventually to running a full ICA session.

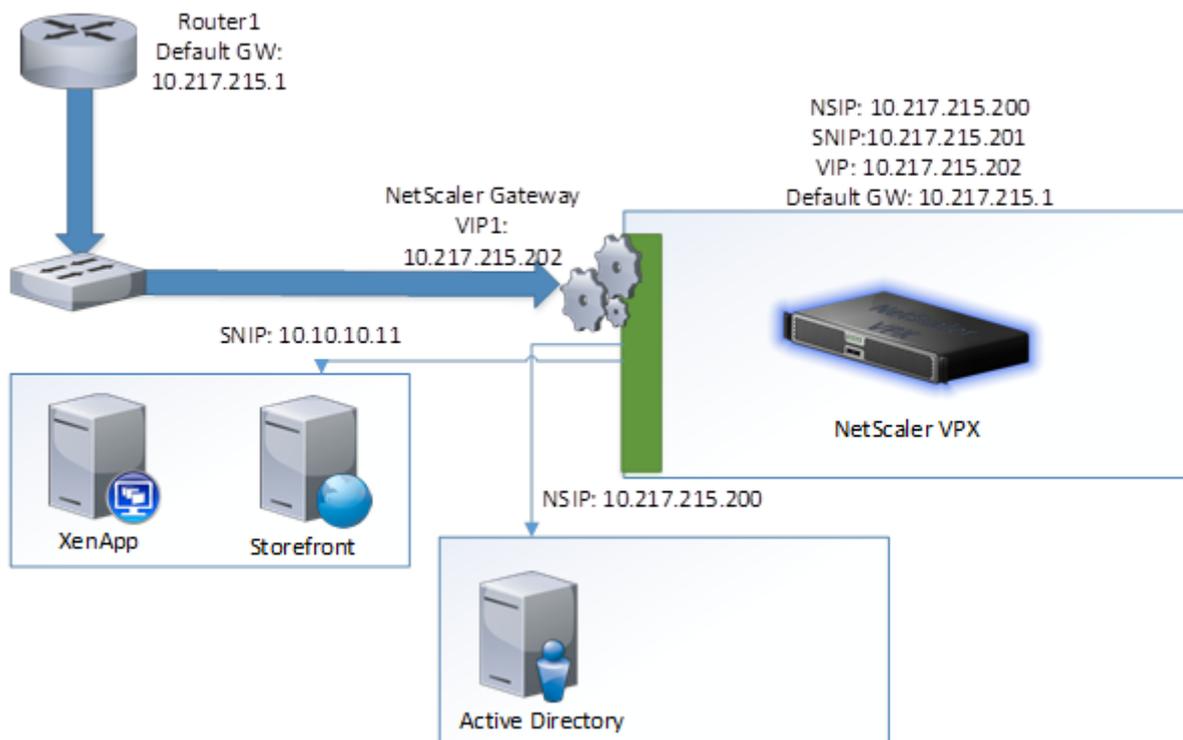
- A user goes to the FQDN of the public IP of the NetScaler Gateway IP

- The user is going to be asked to authenticate based upon the authentication policy.
- After the user has authenticated, NetScaler will assign a session cookie for this user for subsequent client requests
- The user's request will be processed through the session policy assigned to the NetScaler Gateway Virtual server, which will check if user is allowed access and what kind of settings are applied
- User will be forwarded to the Storefront Server Web interface using the SNIP address of the NetScaler
- Storefront will accept the connection because the address it is coming from is the same which is specified in the appliance list on Storefront
- Authentication credential are forwarded from the NetScaler to Storefront based upon settings in the session policy
- Storefront will validate the user by doing a callback to the NetScaler Gateway virtual server on which the user authenticated
- Storefront forwards the credentials to the delivery controllers specified in the Citrix farm to get a list of available resources for that particular user
- Delivery Controller will return with a secure ticket and a list of available resources from that XenDesktop farm
- Storefront generates a list of resources available for that user in the web page or using Receiver
- The user clicks on a resource. The request is sent to the NetScaler Gateway which then forward to the Storefront server with the Application name
- Storefront queries the delivery controllers for the system to connect to
- Delivery Controller responds if resources are available for connection.
- Storefront generates an ICA file which contains information of the NetScaler Gateway so Citrix Receiver knows where to connect to and an STA secure ticket
- Citrix Receiver makes a connection to the NetScaler Gateway address specified in the ICA file
- NetScaler Gateway Virtual Server asks the STA server stored in the ICA file if it is valid and asks to connect
- The STA server validates the request and return information on where to find the VDA agent for that particular resources

Now the connection is up and running where resources are being proxied using TCP:443 between the end-user and the NetScaler Gateway and using TCP 1494/2598 between the SNIP and the VDA agents

Virtual Server setup

For this first setup, we are going to start with a simple one-armed mode, where we only have one NIC connected. In this scenario we have all three IPs in the same subnet, where we have some form of NAT in front where we have a public IP-address, which resolves to our internal VIP. This setup is going to be using ICA-proxy which means that it will only be accessible using Citrix Receiver.



This is a simple deployment which can easily be understood using the traffic flow chart.

Go into the NetScaler management portal, go down into **NetScaler Gateway** right click and choose **enable feature**. Go into **Virtual Servers** and click **Add**

Define a name and IP-address

Name*
NSGW

IP Address Type*
IP Address

IPAddress*
10 . 217 . 215 . 202 IPv6

Port*
443

Click **More** button and choose **ICA-only**



Now there are more settings here as well, and some of them will be covered later in this eBook but to give a short summary of what they all do I described them below.

- **RDP Server Profile:** This allows the NetScaler virtual server to act as an RDS Gateway and proxy connection to RDP servers.
- **Maximum Users:** Here, we can specify how many concurrent users are allowed to log in to the virtual server, zero means unlimited.
- **Max login attempts:** Here, we specify how many failed logins a user can have against a virtual server.
- **Failed login timeout:** This specifies how long a user is locked out after exceeding the maximum login attempts.
- **Windows EPA Plugin Upgrade:** If the virtual server should do a plugin upgrade for the EPA clients for Windows
- **Linux EPA Plugin Upgrade:** If the virtual server should perform a plugin upgrade for the EPA clients for Linux.
- **Mac EPA Plugin Upgrade:** If the virtual server should perform a plugin upgrade for the EPA clients for Mac.
- **Login Once:** This feature enables seamless SSO for the virtual server, which eliminates the need for users to re-authenticate their credentials by, for instance, going from an SSL VPN-based virtual server to an ICA Proxy virtual server.

- **Double-hop:** This is needed if we have a double DMZ and the network traffic needs to traverse between two NetScaler appliances.
- **DTLS:** DTLS is a derivation of the SSL protocol that provides security services, but it is built upon the UDP protocol. This feature is used for services such as audio using UDP for ICA Proxy and Framehawk
- **ICA Proxy Session Migration:** This feature is used to migrate an existing session from a user; when the user changes the device, their session is then migrated to that new device. This ensures that a user can only have one session at a time.
- **Enable Device Certificate:** This feature specifies if the virtual server should check for device certificates as part of the Endpoint scan.
- **Enable Authentication:** This specifies if the virtual server should authenticate. This will then trigger it to present a username and password dialog box to users trying to connect. If this is not enabled and the virtual server is set to ICA Proxy, the users will be redirected to the web interface address.
- **Down State Flush:** Enabling this feature will allow the virtual server to flush all new and existing connections if the virtual server is set to disabled. If this feature is not enabled and we disable the virtual server, all existing connections will be honored, but no new connections will occur.
- **Appflow Logging:** This feature is used to send AppFlow data to a collector, which is used for HDX data. This might be, for instance, Citrix NetScaler Insight or for a third party such as Goliath IT analytics for NetScaler.

Certificates

Click OK. Next we need to add a digital certificate since a NetScaler gateway cannot operate without one. For testing purposes we can proceed with a self-signed certificate but I highly suggest that you use a third-party signed digital certificate like ones from GoDaddy or DigiCert.

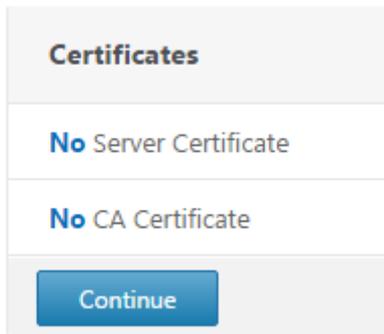
In order to create a certificate, you first need a private key and a **Certificate Signing Request (CSR)**, which is sent to a third-party certificate issuer such as Go Daddy. In NetScaler, go to the **SSL** pane under **Traffic Management** in the GUI from here we can create a custom private key and a CSR which we then can send to our CA issuer. This menu also allows you to import the certificate itself, along with the intermediate and root certificates. It also has an option to import PFX

certificates, which often bundle the private key, the intermediate certificate, and the PFX certificate itself in one file. You can also use OpenSSL from NetScaler to convert a CRT certificate to a PFX, using the guide located at <http://bit.ly/1lzMvEq>

For this example, I have an internal PKI based upon Active Directory Certificate Services which I used to issue myself a certificate to the FQDN **nsgw.test.local**

Note: this is just a random domain name, the certificate name should reflect the external FQDN you want to use

Click on the server certificate pane



Click on the **+ Sign** to add a certificate. The certificate that either be placed on a share which the end-client we are using has access to or already placed under the ns/ssl/ folder on the NetScaler. The most important part is that we have the private key present in the Certificate, unless it won't operate. The reason for that is when a user opens up a connection to the NetScaler Gateway, they will generate a public key. This key is used to encrypt data going to the Gateway and only the Private key can open that traffic. So for each client connecting will get its own public key and will only be able to communicate encrypted with the one with the private key. When generating a certificate using ADCS you need to allow export of the private key. This is also typically included in a PFX file. So define a key pair name, choose the certificate and the key either locally or on the appliance itself.

Certificate-Key Pair Name*

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

Certificate File Name*

 ▼

Key File Name

 ▼ ?

Certificate Format

PEM DER

Local ✓
Appliance

Password

Certificate Bundle
 Notify When Expires

Notification Period

Define the password for the certificate which will allow us to import it on the appliance. The notification period allows the NetScaler to generate an alert when the certificate is about to expire, the date here can be changed from 10 to 100 days if required.

Click **Install** and then **Bind** to bind the certificate to the virtual server. We will come back to more of the SSL configuration later in the eBook.

Authentication

Now click continue, now we need to define an Authentication point. This might consist of two-factor authentication mechanisms like LDAP + RADIUS for the simplicity this first part will only cover LDAP. Click on the **+ Sign** under Authentication

NOTE: Additional authentication features are covered in the authentication module in this book.

Choose LDAP and Primary

Choose Policy*

 ▼

Choose Type*

 ▼

Click **Continue** then click **+ sign** on the select policy to create a new policy, leave the priority at the default value

Now we need to define an expression. Expressions are rules which needs to be evaluated before a policy can be processed. In this example we are using the global expression *ns_true* which translates into “all traffic on the NetScaler” which means that this policy is going to be processed for everyone who wants to access this virtual server.

Define a **name for the policy** enter the expression “*ns_true*”

Click the **+ Sign** under server to define a new LDAP server. Here we need to define an IP-address or hostname for an Active Directory server. We also have the option to choose between PLAINTEXT or SSL/TLS. SSL/TLS gives us the option to have password change capabilities on the NetScaler Gateway but this will be covered later in the eBook.

Note: that the default option is port 389. However, we can use NetScaler in conjunction with cross-domains within the same forest as long as you point the LDAP policy to the Global Catalog port (3268), since the Global Catalog servers are the only ones that know of objects in other domains

Name*
AD.test.local

Server Name Server IP

IP Address*
10 . 217 . 215 . 101 IPv6

Security Type*
PLAINTEXT ?

Port*
389

Server Type*
AD

Time-out (seconds)
3

Authentication

Connection Settings

Base DN (location of users)
CN=users,DC=test,DC=local

Administrator Bind DN
administrator@test.local

BindDN Password
Administrator Password
.....

Confirm Administrator Password
.....

[Retrieve Attributes](#)

So define a baseDN (Which OU does our users reside in) If we have a domain name called test.local and we have a OU called users the BaseDN would look like this

CN=users,DC=test,DC=local, we also need to define a service account which the NetScaler is going to be used to do LDAP bind to the domain. The service account needs to following rights

- Read access to the user objects in the LDAP directory in order to search for user accounts.
- Read access to the Base DN (for example, DC=test, DC=com) with the correct attribute that is used as the LDAP Login Name (for example, samAccountName).

After we have defined a Password, we can click the **Retrieve Attributes** button, this is going to initiate a connection from the end-user client where the browser is running to verify a successful connection to the LDAP server. After the policy has been defined, it is going to be using the NSIP to talk with LDAP.

Next, we need to define which attributes that should be used for authentication. Here we have the Server Logon Name attribute which should be defined to sAMAccountName which allows users to logon using their default usernames. We could also use UserPrincipalName this would allow users to logon using their e-mail addresses, for instance msandbu@test.local

One thing to remember however is that Storefront does not like UPN based logon, so in case we want to use that we need to change the SSO Name attribute to be samaccountname like in the screenshot below.

The screenshot shows a configuration window titled "Other Settings". It contains the following fields:

- Server Logon Name Attribute:** A dropdown menu showing "--< New >--" and a text input field containing "userPrincipalName".
- Search Filter:** An empty text input field.
- Group Attribute:** A dropdown menu showing "memberOf".
- Sub Attribute Name:** A dropdown menu showing "cn".
- SSO Name Attribute:** A dropdown menu showing "--< New >--" and a text input field containing "samAccountName".

If we do not require UPN based logon, leave the SSO name attribute field blank and just use sAMAccountname as server logon name attribute.

We also need to define the Group Attribute **memberOf** and **Sub attribute name, which is going to be used when we define authentication**, based upon Active Directory Group memberships.

After we are done with the configuration, we can click Create, Create and OK to bind the policy to the virtual server.

NOTE: There are steps to troubleshooting authentication and certificate connections but they are in the troubleshooting section of this ebook.

SSL Settings

By default, the SSL Settings on a NetScaler are not setup for maximum security and with some minor tweaks, we can adjust them.

After we are done with the Authentication policy we get the SSL settings. These settings can be added to a virtual server either by using a central SSL policy or using SSL parameters which are setting for each virtual server. If we use a SSL policy, we will disable the SSL parameters menu option.

NOTE: For many the goal is to achieve an A+ on SSLlabs.com for the external virtual server, the recipe for that can be found here → <http://bit.ly/22dmSP2> it consists of enabling only TLS 1.2, using HSTS and using specific ciphers and enabling features like deny renegotiation. This is also covered in the SSL section of this eBook and not covered in the section.

Profiles

After configuring the SSL settings, we have the profiles option available. There are three different profiles from the menu.

- Net Profiles (Used to enforce the use of a SNIP for backend connections to XenDesktop and Storefront)
- HTTP Profiles (Used to define how HTTP should behave, enable use of HTTP/2, SPDY and so on)
- TCP Profiles (Used to define how TCP should behave, use of congestion algorithms and so on)

All these different profiles and different options are available under System → Profiles. If we have a NetScaler with multiple SNIPs defined within the same subnet, we can create our own Net Profile, which contains the SNIP we want to have, and then bind that net profile to this virtual server. This will enforce the Virtual Server to use that particular SNIP for backend communication.

The most important part of the profiles is the TCP profiles. By default, it uses the `nstcp_default_profile`, which has not been adjusted for a long time, and is aimed at multiple purposes TCP services.

This should be changed to `nstcp_default_XA_XD_profile`, this TCP profile is tuned specifically for ICA sessions which includes use of TCP settings like SACK, Window Scaling, Nagle and so on.



The screenshot shows the 'Profiles' configuration page in a NetScaler web interface. It is divided into two main sections: 'Net Profile' and 'HTTP Profile'. The 'Net Profile' section has a dropdown menu set to 'SNIP' with '+' and edit icons. The 'TCP Profile' section has a dropdown menu set to 'nstcp_default_XA_XD_profile' with '+' and edit icons. The 'HTTP Profile' section has a dropdown menu set to 'nshttp_default_profile' with '+' and edit icons. At the bottom left, there is a blue 'OK' button.

TCP Profiles

As mentioned there is a default TCP profile on the NetScaler called *nstcp_default_profile* which is applied on all TCP connections. This section contains more detailed information on how we can properly tune TCP parameters depending on different criteria.

Now TCP has many different parameters that we can define or change, but not in the default TCP Profile. This is because NetScaler is by default configured to be able to fit into most environments and most networks. Many of the TCP options might give a performance boost, or they might also degrade performance if not properly configured or if you are unsure about the different options.

All other TCP settings should be configured in TCP Profiles so you can apply different TCP settings to different virtual server and different services. For instance, TCP traffic should behave differently for a service containing a website for mobile devices, then for a backend service located on a fast Ethernet LAN connection.

A couple settings should be enabled in **nstcp_default_profile** on every NetScaler. These settings are disabled by default.

- Window Scaling (leave default scaling factor of 4)
- Selective Acknowledgement

Details about these parameters is described on a later page. By default, NetScaler contains the following built in TCP profiles:

- **nstcp_default_tcp_lfp**: This profile is useful for long fat pipe networks (WAN) on the client side. Long fat pipe networks have long delay, high bandwidth lines with minimal packet drops.
- **nstcp_default_tcp_lnp**: This profile is useful for long narrow pipe networks (WAN) on the client side. Long narrow pipe networks have considerable packet loss occasionally.
- **nstcp_default_tcp_lan**: This profile is useful for backend server connections, where these servers reside on the same LAN as a NetScaler appliance.
- **nstcp_default_tcp_lfp_thin_stream**: This profile is similar to the **nstcp_default_tcp_lfp** profile. However, the settings are tuned for small packet flows.
- **nstcp_default_tcp_lnp_thin_stream**: This profile is similar to the **nstcp_default_tcp_lnp** profile. However, the settings are tuned for small packet flows.

- **nstcp_default_tcp_lan_thin_stream**: This profile is similar to the `nstcp_default_tcp_lan` profile. However, the settings are tuned to small packet flows.
- **nstcp_default_tcp_interactive_stream**: This profile is similar to the `nstcp_default_tcp_lan` profile. However, it has a reduced delayed ACK timer and ACK on PUSH packet settings.
- **nstcp_internal_apps**: This profile is useful for internal applications on a NetScaler appliance. This contains tuned window scaling and SACK options for the required applications. This profile should not be bound to services other than internal services.
- **nstcp_default_XA_XD_profile**: This profile is aimed specifically for ICA connections and should only be used on the Netscaler Gateway vServer.
- **nstcp_default_mobile** : This profile is aimed at mobile service and uses another congestion algorithm which is specifically aimed at mobile connections using 4G and such.

Now the principle behind the built-in TCP profiles is that we do not need a deep understanding of TCP and the different parameters, and that we can just assign one of the built-in profiles to a service based on what kind of network we have.

On the other hand, if we want to customize our own TCP profile we need to know the different attributes within the profile. Within a profile, we have the following attributes:

- **Window Scaling** is a TCP option that allows the receiving point to accept more data than allowed in the TCP RFC for window size before getting an acknowledgement. By default, the window size is set to accept 65,536 bytes. With Window Scaling enabled, it bitwise-shifts the window size so the window size is increased. This is an option that needs to be enabled on both endpoints in order to be used, and will only be sent in the initial three-way handshake.
- **Selective Acknowledgement (SACK)** is a TCP option that allows for better handling of TCP retransmission. In the scenario where two hosts communicate with SACK not enabled and suddenly the receiver does not receive some out of order packets, then the receiver acknowledges the last in-order packet it received and the sender has to resend all later packets, even if they've already been received (out of order). With SACK enabled, the receiver will notify the sender of only the specific packets it is missing. This allows for faster communication recovery since the sender does not need to resend all the packets.

- **Forward Acknowledgement (FACK)** is a TCP option that works in conjunction with SACK and helps avoid TCP congestion by measuring the total number of data bytes outstanding in the network. Using the information from SACK it can more precisely calculate how much data it can retransmit.
- **Nagle's algorithm** is a TCP feature that tries to cope with small packet problems. Applications such as Telnet often send each keystroke within its own packet, creating multiple small packets containing only 1 byte of data, which results in a 41-byte packet for one keystroke. The algorithm works by combining a number of small outgoing messages into the same message, thus avoiding overhead. ICA is a protocol that operates by sending many small packets, which might create congestion on the network; this is why Nagle is enabled in the *nstcp_default_XA_XD_profile*. In addition, since many might be connecting using 3G or Wi-Fi, which might in some cases, be unreliable when it comes to changing channel, we need options that require the clients to be able to reestablish a connection quickly and that allow the use of SACK and FACK.

NOTE: Nagle might have negative performance on applications that have their own buffering mechanism and operate inside the LAN. Since ICA-proxy mostly uses TCP, except for Framhawk traffic using DTLS, this should be enabled for NetScaler Gateway vServers. If we take a look at another profile such as *nstcp_default_lan*, we can see that FACK is disabled; this is because the resources needed to calculate the amount of outstanding data in a high-speed network might be too much for the CPU to handle.

- **Maximum Burst Limit:** This setting controls the burst of TCP segments on the wire in a single attempt. A higher limit here ensures faster delivery of data in a congestion-free network. Limiting bursts of packets helps to avoid congestion.
- **Initial Congestion Window size:** Initial congestion window defines the number of bytes that can be outstanding in the beginning of a transaction. The default size is 4 (that is 4*MSS).
- **TCP Delayed ACK Time-out (msec):** To minimize the number of ACK packets on the wire, this NetScaler feature by default sends ACK only to a sending node if the NetScaler receives two data packets consecutively or the timer expires; the default timeout is 200 ms.
- **Maximum ooo packet queue size:** This feature allows out-of-order packets in TCP streams to be cached in system memory and reassembled before NetScaler processes them. By default, the value is 64; we can define a value of 0 that means no limit but this will put a lot of strain on the NetScaler memory usage.

- **MSS and Maximum Packets per MSS:** With this setting, we can specify the maximum segment size to be used for TCP transactions. It is important to note that jumbo frames will override this setting, since MSS is TCP sizes and MTU is IP packets, which is lower in the network layer. Therefore, if we specify a higher MTU size on the interface by enabling jumbo frames the MSS size will increase as well.
- **Maximum Packets Per Retransmission:** This setting controls how many packets are retransmitted in a single attempt. This is used with TCP Reno that used a partial ACK value to notify NetScaler that it has received some of the packets but not all.
- **TCP Buffer Size (bytes):** The buffer size is the receiver buffer size on the NetScaler. The buffer size is advertised to clients and servers from NetScaler and it controls their ability to send data to NetScaler. The default size is 8K and in most cases, it will be beneficial to increment this when communicating with internal server farms.

NOTE: There is a good example of tuning of TCP settings from AOL located here on Slideshare → <http://www.slideshare.net/masonke/net-scaler-tcpperformancetuningintheaolnetwork>. Also see Citrix Knowledgebase article CTX121149 *Recommended Settings and Best Practices for Generic Implementation of a NetScaler Appliance* at <http://support.citrix.com/article/CTX121149>

Another important configuration piece in the TCP profile is the congestion algorithm, which specifies how the traffic flow should react when congestion occurs.

The following congestion algorithms are available in NetScaler:

- Default (based upon TCP Reno)
- Westwood (based upon TCP Westwood+)
 - BIC
 - CUBIC
- Nile (based upon TCP-Illinois)

Now this chart below might give you a good indication on what to use what.

New Reno	Westwood+	BIC	CUBIC	Nile
<ul style="list-style-type: none"> • Slight modification over TCP Reno • Able to detect multiple packet loss and thus much more efficient than Reno • Takes one RTT to detect packet loss 	<ul style="list-style-type: none"> • Sender side modification of window algorithm of TCP Reno • Bandwidth estimation to detect packet loss • Intended to better handle large pipes 	<ul style="list-style-type: none"> • Optimized for high speed long distance networks • Algorithm tries to find where to keep the window size at for a long period of time • Not good for low RTT and low speed environment 	<ul style="list-style-type: none"> • Designed to simplify window adjustment of BIC • Window growth is independent of RTT, but dependent on last congestion event • More performant and stable compared to BIC 	<ul style="list-style-type: none"> • Citrix proprietary algorithm based on TCP Illinois, suitable for high speed networks like LTE, LTE advanced and 3G • Achieves higher throughput than standard TCP algorithms

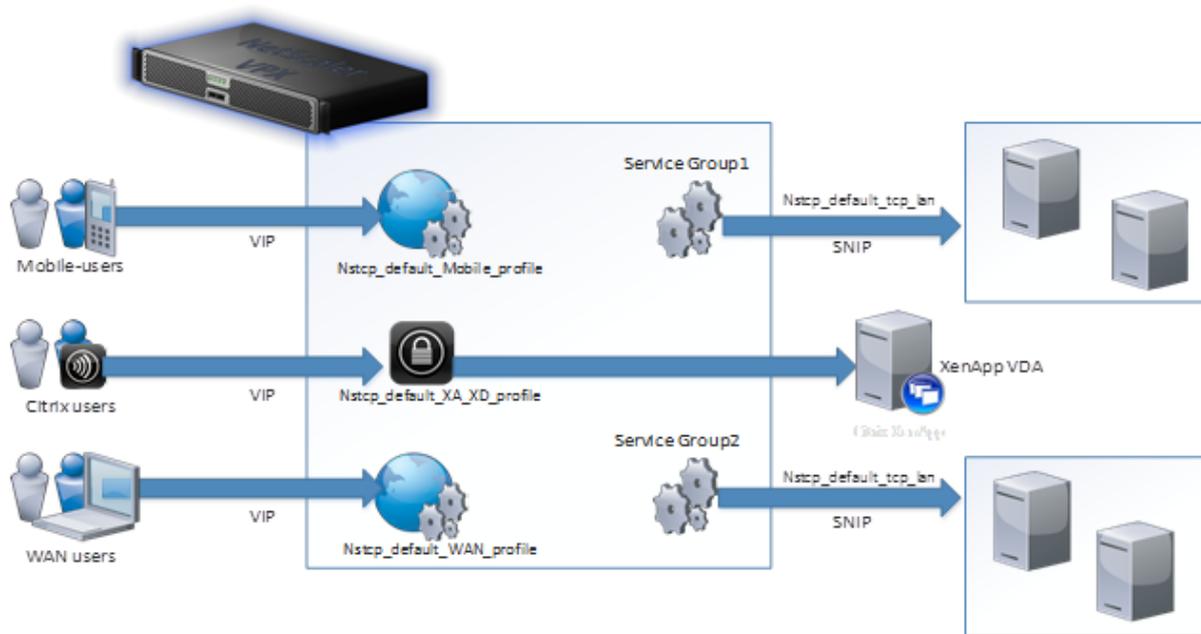
Source: citrix.com

Westwood is aimed at 3G/4G connections, or other slow wireless connections. BIC is aimed at high-bandwidth connections with high latency, such as WAN connections. CUBIC is almost like BIC but not as aggressive when it comes to fast-ramp and retransmissions. Note that CUBIC is the default TCP algorithm in Linux kernels from 2.6.19 to 3.1. Nile is a new algorithm created by Citrix and was introduced in NetScaler 11, which is based upon TCP-Illinois. This algorithm is targeted at high-speed, long-distance networks. It achieves higher throughput than standard TCP and is also compatible with standard TCP.

Therefore, now we can choose that congestion algorithm which is better suited for particular services. For instance, if we have a service that serves content to mobile devices, we could use the `nstcp_default_mobile` TCP profile, which uses the Westwood congestion algorithm. Westwood avoids working on the fixed parameters where we cut the congestion window by half, rather it follows the heuristic model to track the estimated bandwidth on client side and accordingly updates the congestion window

TCP Profiles as mentioned can be attached to a virtual server (incoming traffic), and to a service or service group (backend traffic) expect for a Netscaler Gateway vServer where we can only specify a TCP Profile to the virtual server, since there is no service attached to a Gateway vServer.

The picture below, shows some examples of how we can use TCP profiles for different services



Published Applications

After we have configured the different profiles we need to add an STA server which NetScaler can communicate with to generate an STA ticket. This can be added under Published Applications → STA Servers → Add binding → Enter the FQDN of the DDC and choose IPv4

STA Server Binding

Secure Ticket Authority Server*

Secure Ticket Authority Server Address Type

Then click Bind. When you are back to the VPN Virtual Server STA Server Binding menu, make sure that NetScaler can communicate with the STA server. It should show an Auth ID if it can communicate. Make sure that we have at least two STA servers added, this is required for session reliability.

Policies

The last part we need to configure is session policies, which define how the virtual server should connect with the backend resources like Storefront and so on. The common practice is to have a policy aimed for Web Receiver (Connections going using a web-browsers) and a policy for native Receiver clients. When a user connects to the virtual server, the server is going to look at the HTTP request and see what kind of agent is connecting and then apply a policy based upon the agent.

So under Policies click the + sign, choose Session and Request policy and click Continue. Click add binding, give it a priority of 100 and click the + sign behind select policy.

Under expressions we have two default expressions that are used.

REQ.HTTP.HEADER User-Agent CONTAINS CitrixReceiver

And

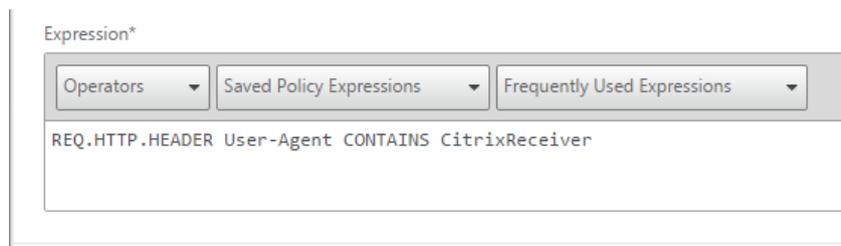
REQ.HTTP.HEADER User-Agent NOTCONTAINS CitrixReceiver && REQ.HTTP.HEADER Referrer EXISTS

So we need to create two sessions, where one is using the first expression and the other is using the second one.

Citrix Receiver policy

Let us start by creating the Citrix Receiver only session policy. We enter the expression in the expression window.

REQ.HTTP.HEADER User-Agent CONTAINS CitrixReceiver



Next we click + sign behind the profile, and this is where we define the ICA settings. For the Citrix Receiver policy, we need to configure the following properties.

Under Client experience we need to define

Clientless Access = Allow

Plug-in Type = Java

Single Sign-on to Web Applications enabled



NOTE: The Image has been cropped to minimize size

Under the **security pane**, we need to define

Default authorization action = Allow

The screenshot shows a configuration pane with three tabs: 'Network Configuration', 'Client Experience', and 'Security'. The 'Security' tab is active. Below the tabs, there is a section labeled 'Override Global'. Under this section, the 'Default Authorization Action*' is set to 'ALLOW' in a dropdown menu, with a checked checkbox to its right.

Under Published Applications pane we need to define

ICA Proxy = ON

Web Interface Address = Define the StoreFront FQDN, which hosts the Receiver for Web site.

Single Sign-on domain = The Local Active Directory domain which we want to use

Account Services Address = This is used for email based discovery which is covered later in this ebook.

The screenshot shows a configuration pane with several settings, each with a checked checkbox to its right:

- ICA Proxy***: Set to 'ON' in a dropdown menu.
- Web Interface Address**: Set to 'http://ddc.test.local' in a text box.
- Web Interface Address Type***: Set to 'IPV4' in a dropdown menu.
- Web Interface Portal Mode***: Set to 'NORMAL' in a dropdown menu.
- Single Sign-on Domain**: Set to 'test.local' in a text box.
- Citrix Receiver Home Page**: An empty text box.
- Account Services Address**: Set to 'http://ddc.test.local' in a text box.

Click OK

Citrix Receiver for Web Policy

Go into policies → Session policies → Add binding, give it the same priority as the other policy → Click the + Sign. Enter the expression

REQ.HTTP.HEADER User-Agent NOTCONTAINS CitrixReceiver && REQ.HTTP.HEADER Referer EXISTS

Give the policy a name and click the + Sign to add a Profile.

The only difference for Citrix Receiver for web is under Published applications

Web Interface Address = Define the entire URL for the Receiver for web site on the Storefront Server.

ICA Proxy*
ON

Web Interface Address
http://ddc.test.local/Citrix/Storeweb

Web Interface Address Type*
IPV4

Web Interface Portal Mode*
NORMAL

Single Sign-on Domain
test.local

Click OK.

We should now have two session policies attached to this virtual server which both have the same priority but have different expressions which needs to be evaluated by the NetScaler when a client connects.

VPN Virtual Server Session Policy Binding			
Priority	Policy Name	Expression	Profile
100	10.217.215.223	REQ.HTTP.HEADER User-Agent CONTAINS CitrixReceiver	10.217.215.223
100	10.217.215.223	REQ.HTTP.HEADER User-Agent NOTCONTAINS CitrixReceiver && REQ.HTTP.HEADER Referer EXISTS	10.217.215.223

Storefront

Finally, we need to define settings on Storefront to allow SSO from NetScaler and define HDX access. This scenario requires that we have a pre-installed Storefront with an existing Store & Web site in our environment.

First thing we need to add is add another authentication method to the Store, this can be done from within the **Storefront console** → **Manage Authentication Methods** and click add on the "Pass-through from NetScaler Gateway

Method	Settings
<input checked="" type="checkbox"/> User name and password i	⚙️ ▼
<input type="checkbox"/> Domain pass-through Can be enabled / disabled separately on Receiver for Web sites	
<input type="checkbox"/> Smart card Can be enabled / disabled separately on Receiver for Web sites	
<input type="checkbox"/> HTTP Basic	⚙️ ▼
<input checked="" type="checkbox"/> Pass-through from NetScaler Gateway	⚙️ ▼

Click OK. Now we need to add a NetScaler Gateway appliance and enable it for remote access. This is basically setting up a trust between Storefront and NetScaler.

Click on the Manage NetScaler Gateways, then click on Add. Enter a display name and FQDN of the NetScaler Gateway virtual server. Choose Authentication and HDX routing option.

Display name:

NetScaler Gateway URL:

Usage or role: i ▼

Click Next, enter STA information (at least two and then enable the Session reliability option) Also important to remember that the STA's we define need to match the ones we defined on the virtual server.

Secure Ticket Authority URLs: i

<http://ddc.test.local/scripts/ctxsta.dll>

<http://ddc2.test.local/scripts/ctxsta.dll>

⬆️ ⬇️

Load balance multiple STA servers

Bypass failed STA for: hours minutes seconds

Enable session reliability i

Request tickets from two STAs, where available i

Click next and under authentication settings define the FQDN of the Callback URL which should be the FQDN of the virtual server. This requires that Storefront can communicate back to the virtual IP.

Version:	<input type="text" value="10.0 (Build 69.4) or later"/>	
VServer IP address: (optional)	<input type="text" value="v10.0: SNIP or MIP, v10.1+: VIP"/>	
Logon type: i	<input type="text" value="Domain"/>	
Smart card fallback:	<input type="text" value="None"/>	
Callback URL: i (optional)	<input type="text" value="https://nsgw.test.local"/>	/CitrixAuthService/AuthService.asmx

Callback is used to verify if a request came from a particular NetScaler appliance, and is also needed for Smart Access. Lastly we need to define remote access for the store. This setting enables Storefront to create ICA files which uses remote access capabilities.

In Storefront Console → Configure Remote Access for this store → Click enable Remote Access for this store and choose setup NO VPN and choose the NetScaler appliance which we previously created in Storefront

Enabling remote access allows users outside the firewall to securely access resources. After you enable remote access, add a NetScaler Gateway appliance.

Enable Remote Access

Select the permitted level of access to internal resources

Allow users to access only resources delivered through StoreFront (No VPN tunnel) **i**

Allow users to access all resources on the internal network (Full VPN tunnel) **i**

Users may require the NetScaler Gateway Plug-in to establish a full VPN tunnel.

NetScaler Gateway appliances:

<input checked="" type="checkbox"/> NSGW i

Add...

Default appliance:

Lastly, we need to configure beacons. Beacons are used to identify if a user is coming externally or internally. You can read the Citrix article on how to set up beacons at <http://support.citrix.com/proddocs/topic/dwsstorefront-21/dws-configure-beacon.html>.

Now, we have successfully set up and configured ICA Proxy.

NOTE: Beacons are only validated for users connecting using Citrix Receiver directly, users that are going via the web receiver will not be probed using beacons and are always treated as external users

Then click OK. We can now verify that the setup works properly. By going to the FQDN of the NetScaler Gateway.

Summary ICA-proxy

To summarize what we need to setup ICA-proxy just by using the steps we went through so far in the book.

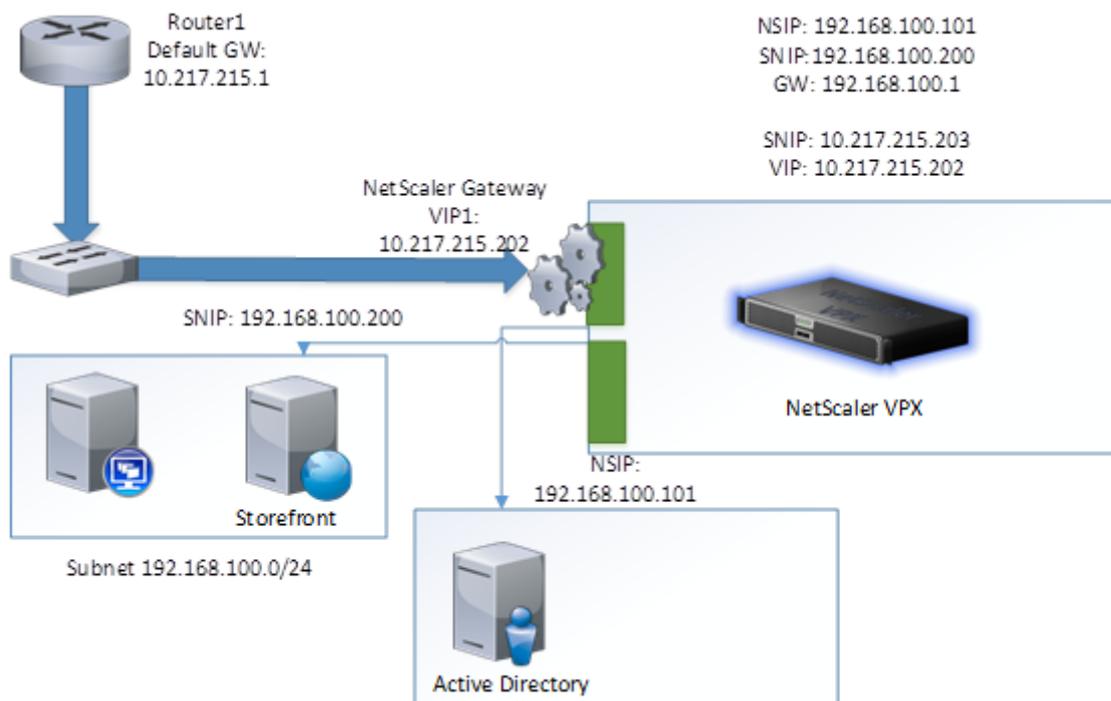
- Setup a NetScaler Gateway Virtual Server with IP address.
- Define ICA-only under General Settings
- Define an authentication policy, in this case we choose Active Directory
- We added a public certificate to validate the authenticity of the Virtual Server
- We configured the virtual server to use a particular TCP profile to ensure optimal traffic flow
- We configured STA servers so that the NetScaler can validate ICA session coming in
- We configured session policies where we defined that the Gateway should operate in ICA-proxy mode and where we defined domain name and the address of Storefront web receiver
- We also defined Storefront settings, where added the NetScaler appliance and defined pass-through authentication and defined the callback mode.

Now if for some reason your ICA sessions are not starting or you are getting an error message.

- Ensure that traffic is flowing directly, check where communication breaks using the traffic flow in the beginning of the section
- Does traffic work from Browser but not from Citrix Receiver?
- Firewall?
- Check if any error messages correlate to the ones in the troubleshooting section in this book

ICA Proxy with two armed

In case we need to have a two-armed mode setup where we have the NetScaler appliance placed in two different network segments we need to take some considerations into the setup. So this example looks like this we have two subnets. The DMZ network is on the 10.217.215.0/24 subnet. Our internal server network is placed on 192.168.100.0/24 network. Our NetScaler is configured with two network interfaces where we have one NIC in the DMZ network and another NIC in our internal server network where our XenDesktop resources are placed.



Here is what is needed to allow it to work in a two-arm configuration

- We need a new SNIP placed in the DMZ network to allow communication back to the endpoints, which connect from the Internet because a VIP on its own do not have packet generating features. If we cannot setup another IP in the DMZ network, we can alternatively enable Mac based forward "MBF" which essentially turns NetScaler into a layer 2 device and focuses on delivering packets where it came from based upon MAC addresses. This can be enabled under System → Configure Modes → MAC Based Forwarding. However, take note that Mac Based Forwarding has some side effects and adds some performance loss.
- In most cases need to re-configure our callback address configured in Storefront, since in most cases an internal server is not allowed to directly communicate back to the virtual server placed in the DMZ zone. In that case, we can to add a **dummy VIP setup on the NetScaler**, which is placed in the internal network. This dummy server just needs an IP-address placed on the server network 192.168.100.0/24 network, have a certificate attached and STA servers defined. After this is configured, we just adjust the Storefront settings to point to this dummy VIP in the callback URL. An example dummy vip looks like this.

VPN Virtual Server

Basic Settings	
Name	NSGW-proxy
IPAddress	10.217.215.228
Port	443
State	Up
RDP Server Profile	-
Login Once	false
Double Hop	true
Down State Flush	false
DTLS	false
AppFlow Logging	false

Certificates
1 Server Certificate
No CA Certificate

Authentication
<i>To add, please click on the + icon</i>

Published Applications
No Next HOP Server
1 STA Server
No Url

Just STA, and the same server certificate as the other netscaler gateway virtual server and with an IP on the internal network which the Storefront server can reach.

- Another possibility is that we need to configure the SNIP to communicate across another subnet in order to be able to communicate with backend resources; in that case, we need to add another static route to the routing table. This can be done under System → Network → Routes → Basic and from there click Add and specify the routing information. For instance, if we have another subnet located on 192.168.200.0 network in that case we just need to point to which gateway the SNIP needs to communicate with in order to ensure traffic flow.

Create Route

Network*

Netmask*

Traffic Domain
 +

NULL Route
 Yes No

Gateway*
 ?

Distance

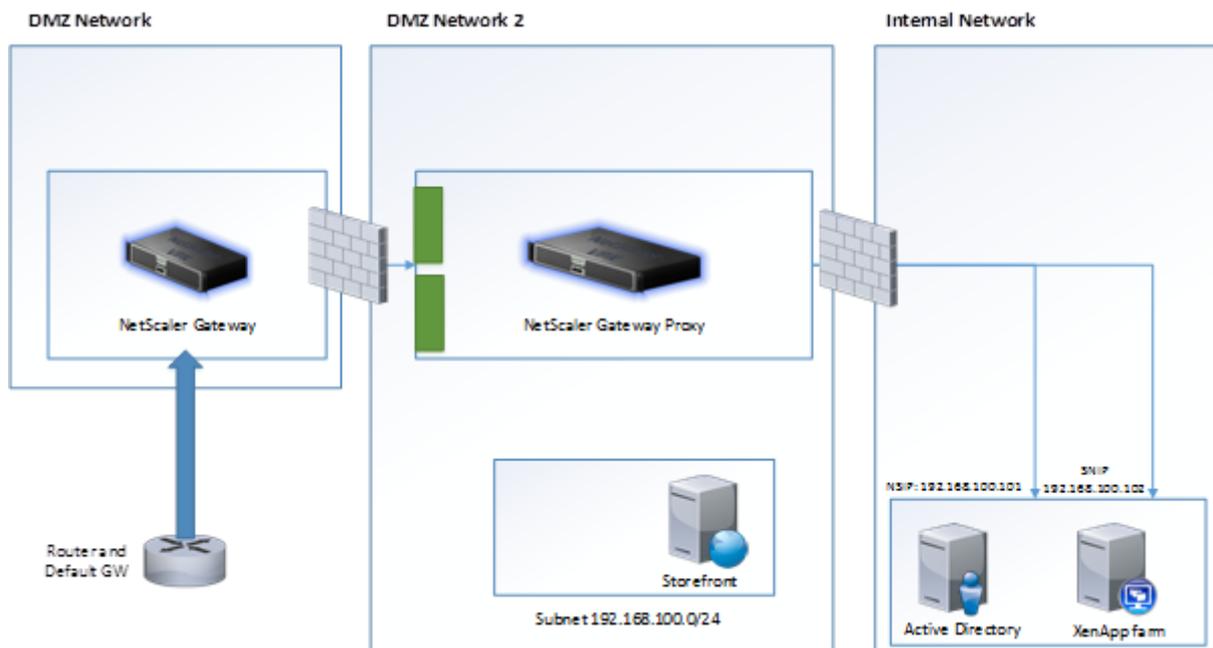
Weight

- Another thing might be that we have a split-DNS scenario where Storefront is never able to get the information about the callback address specified, in those cases it might be beneficial to change the hosts file on Storefront to do a direct mapping between IP and DNS name which is specified in the callback address.

Double-hop configuration

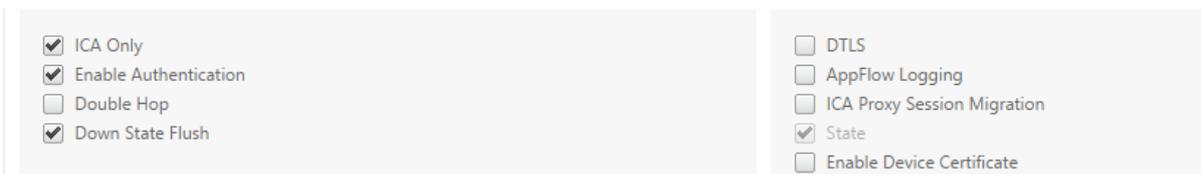
Some organizations have multiple security zones with multiple firewalls to protect their internal network, and to be able to communicate with the internal network a connection needs to traverse between more zone where they typically split DMZ into two stages. This type of network configuration is called double-hop DMZ.

NOTE: Important to remember that this is only supported for ICA-proxy setup and not supported with other type of NetScaler Gateway features like SSL VPN and Full VPN with Endpoint agents.



So on the **NetScaler** in the first DMZ network we need to configure a NetScaler Gateway virtual server

Create a new virtual server with an IP-address and a name, under Basic Settings click more and enable the ICA only feature.



Then click OK, setup all our basic policies like authentication and session policies, like we would in a regular ICA-proxy setup. Lastly we need to define STA connection from the first NetScaler as well, and lastly we define **Next Hop server** which is defined under **Published Applications**. Here we enter the **IP-address of the Virtual Server** that we are going to create on the NetScaler called "**NetScaler Gateway Proxy**" which resides in the other DMZ zone. We can define it using IP or FQDN, we also specify the port which the virtual server is going to communicate with, by default the recommended setting is just using port 443 and setting secure enabled.

Next Hop Servers			
Name	IP Address/FQDN	Port	Secure
NSGW-proxy	10.217.215.228	443	Yes

From there go into the NetScaler Gateway Proxy in the other DMZ, create a new virtual server without any policies, just in plain ICA-only mode and we also need to define under Basic Settings, Enable **Double Hop** and clear the **Enable Authentication** option as well.

<input checked="" type="checkbox"/> ICA Only	<input type="checkbox"/> DTLS
<input type="checkbox"/> Enable Authentication	<input type="checkbox"/> AppFlow Logging
<input checked="" type="checkbox"/> Double Hop	<input type="checkbox"/> ICA Proxy Session Migration
<input type="checkbox"/> Down State Flush	<input checked="" type="checkbox"/> State
	<input type="checkbox"/> Enable Device Certificate

It is also important that both virtual servers have valid certificates and that both virtual servers can trust each other's certificates. Best purpose is to use two different certificates from the same trusted root CA, this ensures that communication works properly.

After these settings are configured, next time when a user's tries to initiate a Citrix Receiver session with the NetScaler Gateway in DMZ Network, the ICA session will be proxied through the NetScaler Gateway Proxy to the backend resources XenApp/XenDesktop. NOTE: that you will not see the ICA session on the NetScaler Gateway Proxy, only way to see sessions that are active on that appliance is by using tcpdump or using the built-in show TCP connections.

Framehawk and Audio over DTLS

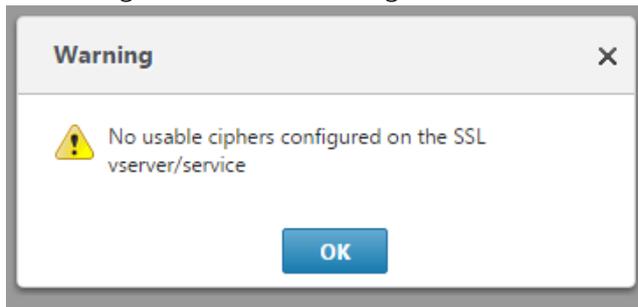
Framehawk is a display protocol which is based DTLS which is an TLS extension on UDP. Also the use of Audio over UDP also uses the DTLS setting. From a NetScaler perspective it is the same configuration, we just need to enable DTLS. The difference is in the XenDesktop environment.

To enable support for DTLS for a NetScaler virtual server, you just need to go into the virtual server → Edit → Basic Settings → Then click enable for DTLS

<input type="checkbox"/> ICA Only	<input checked="" type="checkbox"/> DTLS
<input checked="" type="checkbox"/> Enable Authentication	<input type="checkbox"/> AppFlow Logging
<input type="checkbox"/> Double Hop	<input type="checkbox"/> ICA Proxy Session Migration
<input type="checkbox"/> Down State Flush	<input checked="" type="checkbox"/> State
	<input type="checkbox"/> Enable Device Certificate
	Comments
	<input type="text"/>

After we have enabled DTLS for the virtual server, we need to rebind the certificate to the virtual server. Go into the virtual server → Certificates → Server Certificates → Mark the existing certificate and choose unbind → then choose bind and reattach the certificate.

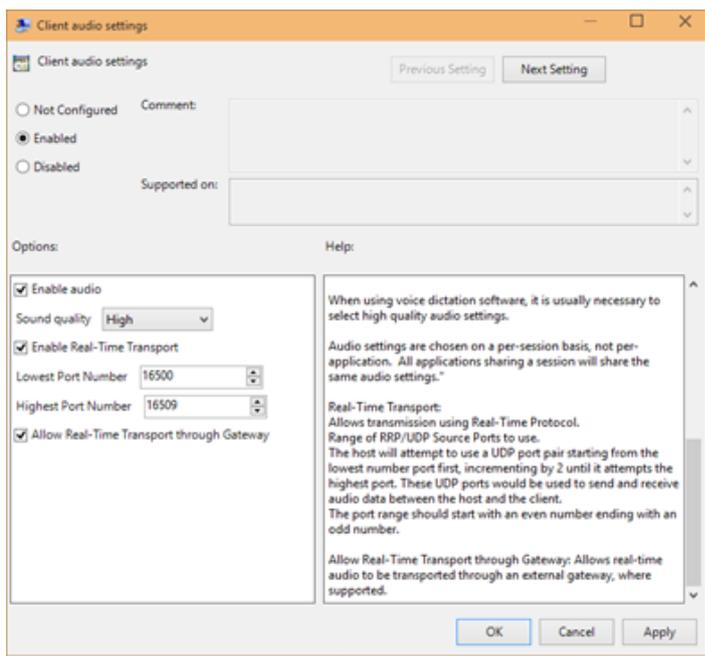
You will get this error message, but this is normal and can be ignored.



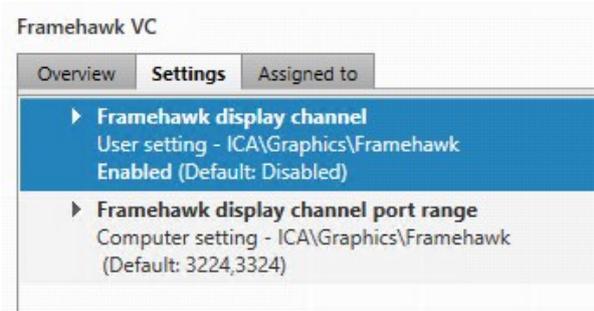
It is important to remember the supported configuration of Framehawk from a NetScaler perspective. This is valid as of now for 11.64 build and my change for future builds.

Scenario	Framehawk Support
NetScaler Gateway	Yes
NetScaler Gateway + GSLB	Yes
Unified Gateway	Yes
HDX Insight	No
NetScaler Gateway IPv6	No
Double hop setup	No
Multiple STA on Gateway	No
NetScaler with High-availability	No
NetScaler with Clustering setup	No

Now lastly if we want to enable Audio over UDP we need to configure some additional properties in the ICA client group policy extension under Computer Configuration → Administrative Templates → Citrix Components → Citrix Receiver → User Experience and under Client Audio settings



And in order to use Framehawk on the VDAs you need to configure the Framehawk Citrix Policies, which can be found inside Citrix Studio.

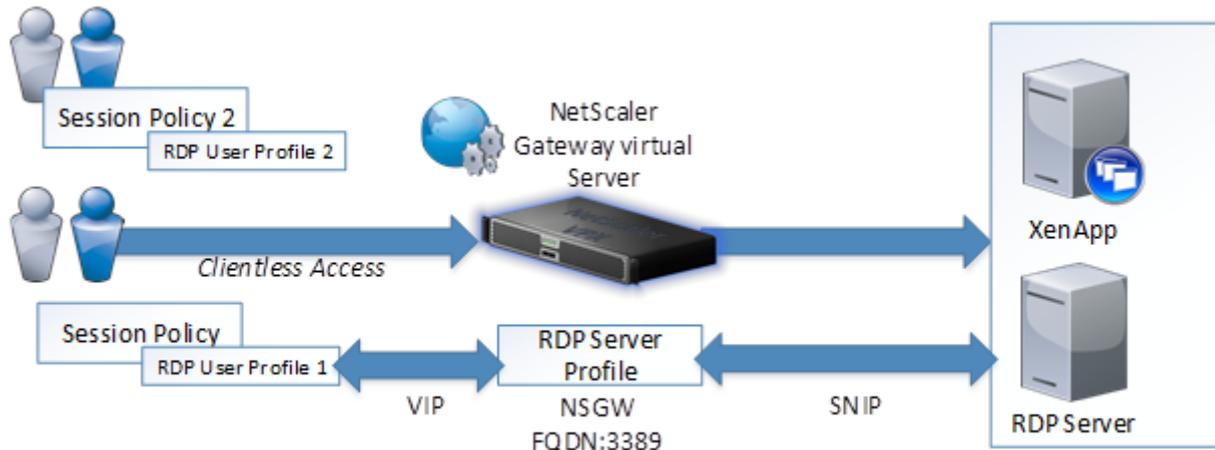


Note: It is important that port UDP 443 is open externally and that UDP ports 3224-3324 are open on the internal firewall if the environment is using the default port ranges.

RDP Proxy

In NetScaler 11 we have the ability to setup the NetScaler Gateway virtual server to act as a Remote Desktop Proxy server. This is done by the use of two parts configuration. One for the server side on the NetScaler gateway where we define an external IP address of the RDP Proxy and port number and shared key. Another part of the configuration is the RDP user profile which is an extension to the session profile and can be bound either globally, to each virtual server or as part of an AAA user or group.

The screenshot below shows the setup and the traffic flow. Note that it requires use of clientless access and will generate an RDP profile which we then can trigger.



NOTE: This feature uses universal licenses for each user also is requires either Enterprise or platinum licenses. We must also make sure that the necessary ports are opened in the firewall. The default port 3389 externally and from the SNIP to the RDSH servers on port 3389

To configure RDS Proxy, first we need to configure the RDS Server Profile which defines which ports and IP address that RDS Proxy server should listen on. This can be configured from NetScaler Gateway → Policies → RDS → Server Profiles. From there click Add

The screenshot shows a web form titled "Create RDP Server Profile". At the top left, there is a "Back" button. The form contains the following fields:

- Name***: A text input field containing "RDSGW".
- RDP IP***: A text input field containing "10 . 200 . 200 . 40".
- RDP Port**: A text input field containing "3389".
- Pre Shared Key***: A text input field containing "qwerty!". To the right of this field is a small question mark icon.

At the bottom of the form, there are two buttons: "Create" (in blue) and "Close" (in grey).

Enter the IP address of the RDS Proxy, this should be the NetScaler Gateway Virtual server IP address. Then define the external port number it should listen to, and lastly define a pre shared key, which is used to do authentication between the client policies and the server profiles.

Now from within the same menu option NetScaler Gateway → Policies → RDS go into Client Profiles and click Add.

From within the client profile, we can define different settings of the RDP connection such as

- Redirecting drives
- Redirecting printers

And so on. The most important part of the client profile settings is further down in the menu.

RDP Cookie Validity (seconds)

60

Add Username In RDP File*

NO

RDP File Name

app.rdp

RDP Host

nsgw.test.local

Multiple Monitor Support*

ENABLE ?

Custom Parameters

Pre Shared Key

qwerty!2

Create Close

- RDS host (Should points at the FQDN of the IP listening on the Server Profile)
- Pre shared key (Should reflect key used in the server profile)

After we have created both a server profile and a client profile, we have to bind them to a virtual server and to a session profile. Go into to NetScaler Gateway → Virtual Server. Click Edit under Basic Settings click the pencil icon.

Choose the pre-created server profile in the list below under RDP server profile. Also make sure that the virtual server is in smart access mode.

RDP Server Profile

RDP ?

Maximum Users

0

Max Login Attempts

Failed Login Timeout

ICA Only

Enable Authentication

Double Hop

Down State Flush

Next we need to bind the RDP client profile to a session policy, which can as mentioned be to a global profile or to an AAA profile.

The screenshot shows a configuration pane with tabs for Network Configuration, Client Experience, Security, Published Applications, and Remote Desktop. The Remote Desktop tab is active. Below the tabs, there is a section labeled "Override Global" containing a dropdown menu for "RDP Client Profile Name" with "RDP" selected, and a checked checkbox.

Make also sure that the session policy has clientless access configured and enabled.

The screenshot shows the "Client Experience" configuration pane. It includes several settings: "Accounting Policy" (dropdown), "Display Home Page" (checkbox), "Home Page" (text input), "URL for Web-Based Email" (text input), "Split Tunnel*" (dropdown set to "OFF"), "Session Time-out (mins)" (text input set to "30"), "Client Idle Time-out (mins)" (text input), "Clientless Access*" (dropdown set to "Allow"), and "Clientless Access URL Encoding*" (dropdown set to "Clear"). Each setting has a corresponding checkbox to its right.

Open up a session profile and go into the RDP pane, specify the pre-created RDP client profile and click OK.

Now there are two ways that we can access and start an RDP connection. The first is either by adding a bookmark resource to the clientless access page, or by using a particular URL.

To add a bookmark resource, go into NetScaler Gateway → Resources → Bookmark

Name*
RDP

Text to display*
DC - RDP

Bookmark*
rdp://dc.test.local

Virtual Server
[Empty]

Icon URL
[Empty] Browse ▼

Application Type
[Empty]

SSO Type
[Empty]

Use NetScaler Gateway As a Reverse Proxy

Comments
[Empty]

Create Close

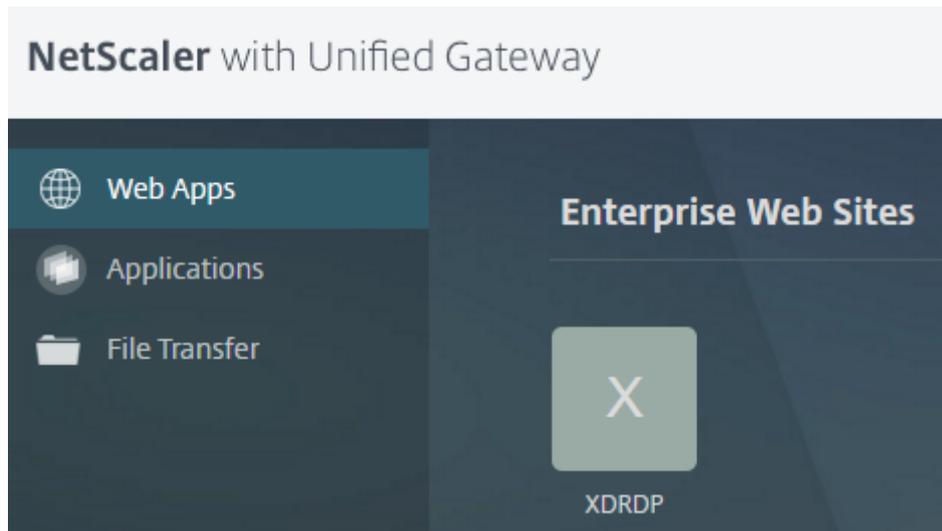
Enter a name and text to display and define a bookmark to point to the internal RDP server we want to access using the form **rdp://fqdn**

And lastly mark the **“Use NetScaler Gateway as a reverse proxy”** then click Create. After we have created the bookmark we have to bind it, either to a virtual server or to an individual AAA group or user. This scenario covers how to add it to a virtual server. Go into NetScaler Gateway → Virtual Server click Edit. Go to published applications and click URL and choose click binding

Bookmarks			
Select Add Edit Delete			
Name	Link	Actual URL	Icon URL
<input checked="" type="radio"/> XDRDP	XDRDP	rdp://dc.test.local	

Mark the pre created bookmark and click select and click bind.

Now that we have added the bookmark, we can go into the FQDN of the NetScaler gateway virtual server and login to the clientless access page.



Note that when we click the RDP icon it will generate a RDP file containing the properties we defined in the client profile settings. It will also attempt to do SSO using CredSSP credentials after we logged in.

NOTE: That if you for instance have domain computers which are not directly accessible using the default username and password which you are logged in with or you want to disable SSO to the RDS Proxy we need to create a traffic policy as such which does that

```
add vpn trafficAction disable-ss0-action http -SSO off -kcdAccount NONE
add vpn trafficPolicy disable-ss0-policy "REQ.HTTP.URL CONTAINS rdpproxy" disable-ss0-action
```

The other option to access RDS hosts is by using an URL after logging into the clientless access page.



[HTTPS://FQDN/RDPproxy/Computername](https://FQDN/RDPproxy/Computername)

Which will then generate an RDP file which can then trigger a connection.

GSLB and Zone feature

This section focuses on delivering GSLB with use of the Zones feature and remote Citrix Access, before we dig into setting up the remote access to Citrix part, I want to cover the basics of GSLB.

GSLB Basics

Global Server Load Balancing (GSLB) allows us to load-balance services across different geographical regions. For instance, large organizations such as Facebook, eBay, and

Microsoft use this technology to load-balance their web services. This might be for proximity reasons, because a user might be redirected to the closest available resource, or to keep redundant services available in case of datacenter failures. For instance, Proximity based DNS is used to determine the closest location available when a user tries to log into Office365.

So the main purposes for GSLB are:

- Performance; user proximity to the closest available resource
- Disaster recovery, where multiple sites can be grouped as primary and standby
- Load balancing resources between different multiple locations

GSLB is based upon the use of DNS. So for instance, when a user wants to go to www.citrix.com, that user's DNS client will do a number of queries against the different DNS servers, until it gets a response from an authoritative source for that domain. An example might be where NetScaler, which is the authoritative name server for that domain, responds with an A-record to that user that is one of the load-balanced virtual servers attached to the domain. Before that record is handed out, NetScaler has done an evaluation of the health state of the different services the user is trying to access, using SNMP based load monitors, **Metric Exchange Protocol (MEP)**, and explicit monitors.

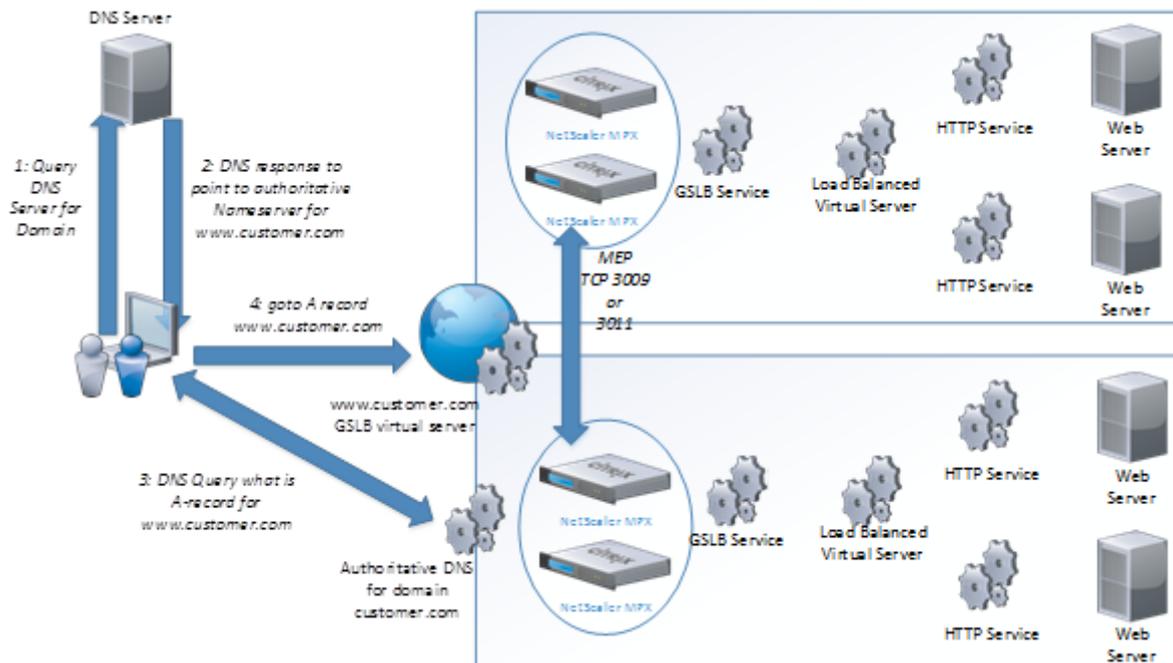
GSLB also consists of different components:

- GSLB sites (that represent a geographical location or datacenter)
- GSLB services (that are linked to a load-balanced vServer)
- GSLB vServers (that consist of multiple GSLB services served from GSLB sites)
- Domains (that on NetScaler are either authoritative or act as a proxy on its behalf)
- MEP used between nodes in each site to exchange information about the state and load on the site
- SSH for GSLB configuration sync

Before we go ahead with talking about design. We need go a bit deeper into MEP.

MEP is a proprietary protocol used by NetScaler to communicate different GSLB site metrics, network metrics and persistence info to the other GSLB sites. Communication with MEP happens on port 3011 or 3009 for secure communication. If we want to use load-balancing methods such as proximity or RTT, we need to have MEP enabled; if not, it will fall back to round robin.

So let us look closer at an example with a small design layout.



So let us image this sample traffic flow. A user wants to go to a website called www.customer.com the endpoint asks its preferred DNS server, which might not have the answer it needs to answer that query. The DNS server responds with the authoritative DNS server for that particular zone customer.com which in this case is an authoritative DNS server service running on one of the NetScaler. Then depending on the GSLB settings will then point, the end user to one of the two GSLB services, which represent a load-balanced virtual server.

Authoritative DNS

As stated we can also use the Netscaler as an authoritative DNS server. This way we can support when setting up GSLB, which we will be going through later in this section. We can add this as a service on the Netscaler by using the following commands.

```
add service SERVICENAME 10.10.10.10 adns 53
```

After we have added this ADNS service on the NetScaler we can quickly verify that it is working by just by adding an A-record to the DNS server and checking from our local client

```
add addrec test.local 192.168.60.77
```

Then we have to do a nslookup from our client using the following parameters

```
nslookup test.local IPADDRESSOFDNS
```

Now we should have one aDNS service running on each site to ensure proper availability and the name servers should be placed as authoritative for the domain we want to use for this feature.

After we have configured the aDNS servers on each site, we can start setting up the GSLB settings.

Zone based GSLB deployment

NOTE: This enhancement is available with NetScaler release 11.0 build 65.x or later, StoreFront release 3.5 or later, and XenApp/XenDesktop release 7.7 or later. With this enhancement, the client IP address is examined when an HTTP request arrives at the NetScaler Gateway appliance, and the real client IP address can be used to create the datacenter preference list that is forwarded to StoreFront.

Configuration of the Zone based GSLB deployment can be found here → <http://bit.ly/1TLSXrO>

VPN and Endpoint analysis

NetScaler Gateway cannot only operate as a Citrix Receiver gateway it can also function as an SSL VPN and or Full VPN solution for endpoints, who needs to connect directly to the corporate network to get access to internal resources. As part of this solution, we can also do extensive checks on the endpoints before they are allowed access to the corporate networks.

So again to setup a VPN NetScaler Gateway solution is consists of multiple steps, the most common ones to setup are a NetScaler Gateway in Smart access mode. Then adding an authentication policy for Active Directory authentication and then adding a session policy which defines how the VPN solution should behave and what kind of VPN features does the user have access to, and what kind of resources should the user have access too. We can also add a pre-authentication policy, which allows us to do a scan of the endpoint before they are allowed to authenticate

We can also bind traffic policies to the virtual gateway server, which can be used to do SSO to backend resources like Outlook Web Access or so on.



So let us start with a simple VPN setup using pre-authentication policies and giving the end-users the option to use a full VPN solution.

Full VPN with endpoint scanning

- Create a NetScaler Gateway virtual server
- Set the virtual server in smart access mode (Remove the ICA-only option)
- Add an authentication policy to the LDAP server
- Under policies choose create new Preauthentication policy
- Under select policy click on the + Sign

Preauthentication policy

Under the expression window here, we have another option called OPSWAT EPA editor. OPSWAT is a security library, which allows us to setup powerful scanning rules on the end clients based upon different parameters.

Name*

Request Action*

Expression* OPSWAT EPA Editor Expression Editor

Operators Saved Policy Expressions Frequently Used Expressions

This is essentially one of the newer security features, which allows us to do endpoint scanning. We also have some features built into the regular expression editor.

Within the regular expression editor, we have some limited options to what we can check.

- General (HTTP, TCP, IP, SSL)
- Client Security (Antivirus, Firewall, Process, Service, File, Registry, Operating System and so on)

- Network based (VLAN,INTERFACE,MAC)
- Date/Time (TIME, DAYOFWEEK, DATE)
- Filesystem (These expressions are most often used in conjunction with fileshares and will be covered later in the eBook, but can be used to detect if a file is present on the end-agent.

Classic expressions can be used for instance if we want to allow/deny access for a particular IP range, or mac address. We can even filter based upon which date or time of the day they are trying to access. For the client security part, we can only check for instance if we have a particular Antivirus vendor installed, it cannot use this to detect if a particular antivirus solution is running, and also we need to define for instance a particular version and a name

Add Expression

Select Expression Type: Client Security ▼

Component
Anti-Virus ▼

Name*
Norton Internet Security

Qualifier
Version ▼

Operator
== ▼

Value*
10.0

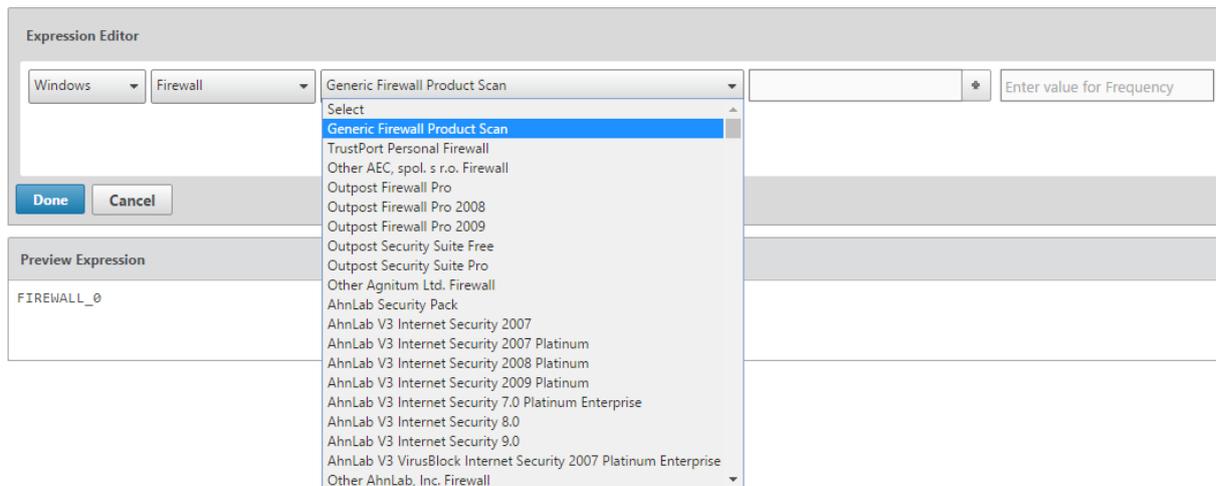
NOTE: However, some options are unavailable here because this is a single scan. So we cannot define a frequency where the EPA agent should scan, this is only an option that is available when defining session policies.

Now if we choose the OPSWAT option on the expression window we have many specific options available, which allows us to do a more thoroughly scan and detection on the endpoint. OPSWAT as mentioned earlier is a library containing a list of most of the different antivirus vendors for instance, they can also do a generic antivirus scan, and they can detect if a particular antivirus is running on not. First of OPSWAT is split into two different main categories which is Mac and Windows, from there we can go into more specific details into what we can choose from

- Antiphishing
- Antispyware
- Antivirus

- Backup Client
- Device Access Control
- Data Loss Prevention
- Desktop Sharing
- Firewall
- Health Agent
- Hard Disk encryption
- Instant Messenger
- Web Browser
- P2P
- Patch Management
- URL Filtering
- MAC Address
- Domain Check
- Numeric Registry
- Non-Numeric Registry

From these categories, we can then go and see if a particular solution is running. So as an example we can do an generic firewall scan on an endpoint.



From the list, choose Generic Firewall Product Scan, then click on the + sign behind the part where we enter the value, this will give us a list of different filter options to choose from.

Within the value option, we choose that the firewall should be enabled and that the authenticity of the product should be evaluated to true.

Create Product Scans		
Version	<	
Authentic	==	TRUE
Enabled	==	TRUE
Comment	==	Generic Firewall Product Scan

OK Cancel

After we have defined the correct value for the policy, click OK and Done. Then we are back to the Create Preauthentication policy window, from there click on the + sign under the request action. Here we need to define the action is taken if the policy evaluates to be true.

Create Preauthentication Profile

Name*
GENERICFWENABLED_TRUE

Action*
ALLOW

Processes to be cancelled

Files to be deleted

Default EPA Group

Create Close

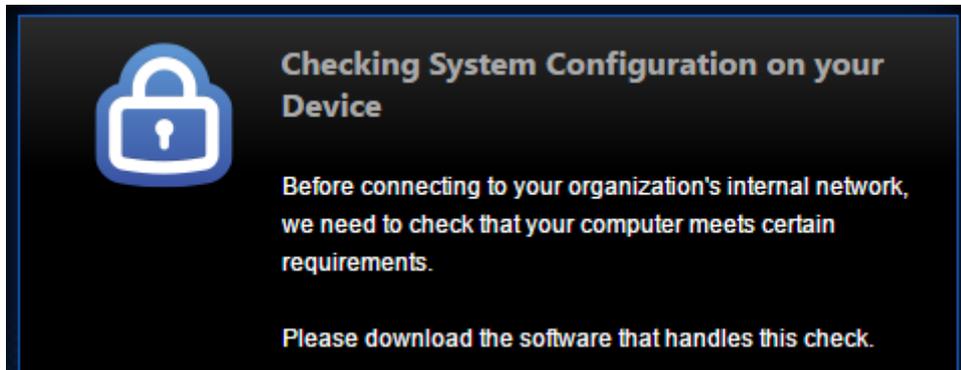
So give it a name and define the action ALLOW, then click CREATE. Now back to the policy creation window, give it a name and click CREATE and click BIND, now we have created a preauthentication policy and bound it to the virtual server.

The flow of this policy will act like this.

- User Connects to the NetScaler Gateway virtual server
- NetScaler Gateway detects that there is a preauthentication policy and notifies the client
- The client needs to download an endpoint agent and run the endpoint scan
- The agent runs the tests predefined in the policy (GENERIC FIREWALL, Authentic? Enabled?)
- If policy evaluates to true, the status is returned back to the virtual server
- Virtual server gives the endpoint access and the client is allowed to authenticate to the virtual server for further access.

NOTE: The endpoint client will store logs under %USERS%\AppData\Citrix\AGEE\nsepa.txt logs. For instance if you are trying this with a self-signed certificate the EPA scan will not start properly.

So when end-users are trying to connect the the virtual server, they will be presented with this screen from the web portal



Which requires the user to download the EPA agent, which will then ask for permission to run.

Windows Update EPA Check

So as an example let us setup a Windows Update Check EPA Scan which will scan the endpoint for a particular patch before the user is allowed in.

First we create an Preauthentication Profile. Go to NetScaler Gateway -> Policies -> Preauthentication Profiles -> Add

Create Preauthentication Profile

Name*
 ?

Action*

Processes to be cancelled
 ?

Files to be deleted

Default EPA Group

Next we need to create an preauthentication policy, Go into NetScaler Gateway -> Policies -> Preauthentication Policies -> Add

From the expression window choose the OPSWAT EPA Editor. Choose Windows, Patch Management and then choose Microsoft Windows Update Agent.

Expression Editor

Windows Patch Management Microsoft Windows Update Agent

Done Cancel

Preview Expression *...Read Only...*

PATCH_80400

Then click on the + sign and enter a version of the WUA

Create Product Scans

Version	<	8.1.11
Comment	==	Microsoft Windows Update Agent

From there click OK and OK until we are back to the menu option where we can choose expression.

Request Action*

preauth_check +

Expression*

Operators Saved Policy Expressions Frequently Used Expressions

CLIENT.APPLICATION(PATCH_80400_VERSION <_ 8.1.11[COMMENT: Microsoft Windows Update Agent]) EXISTS

Now here we can alter the PATCH number which just by default is set to 80400, but that number can be altered directly to be another patch which is installed on the end-user device.

NOTE: The number version of WUA can be found locally on an endpoint, more info here → [https://msdn.microsoft.com/en-us/library/windows/desktop/aa385815\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa385815(v=vs.85).aspx)

After we have altered the expression, we can click create.

Lastly we have to bind this preauthentication policy to the virtual server. Go into the NetScaler Gateway Virtual Server → Policies → Preauthentication Policy and under select policy choose "Click to select" and choose the precreated policy.

Policy Binding

Select Policy*

PREAUTH
>
+
✎

► More

Binding Details

Priority*

100

Bind
Close

Click Bind and Done.

Now that we are done with the preauthentication policy and the authentication policy, we need to define a session policy, which is used to define how the agent should behave for the end-user after successfully authenticating.

Session policy

Under the virtual server click into policy then click the + sign to add another policy. From the list choose session and click OK.

Click Add binding and under select policy click on the + sign to add another policy. Under the expression field here, we are again presented with the option to do a classic expression, which allows us to check the file system, client settings and so on. Since this is a session policy, we have some other options available when defining the client security settings under the classic expressions.

For instance, we now have the option to specify a frequency to when it should to another check on the endpoint. This means it will be a reoccurring check on the endpoint. If we for instance do a file system check for a particular file, the file is removed, and a new EPA scan detects that the file is gone the VPN session will be terminated. Using the OPSWAT editor, we have the same options as we had with the Preauthentication OPSWAT policies.

For an example, we can use the ns_true policy, which defines that all settings in this session policy should be applied to all connections. After we have defined an expression click the + sign under Profile to add a Session Profile.

NOTE: This session policy will force all network traffic to go through the VPN tunnel, this is part of the setting when we define Split-tunneling mode to off. Also by default, the source traffic will be initiated from the SNIP, which is available.

As an example session profile for full VPN access, there are just some settings that needs to be defined

Client Experience

- Home page
- Split tunneling (Should be set to OFF)
- Clientless Access (Should be set to OFF)
- Plug-in type (Should be set to Windows/Mac)
- Single-sign on to web applications (Should be defined as enabled)

Security

- Default Authorization Action (Should be set to ALLOW)

Published Applications

- ICA Proxy (Should be set to OFF)

Split tunneling

As mentioned, that by default when setting up NetScaler VPN all traffic will be routed via the VPN tunnel. We have an option called Split tunneling which allows us to define which traffic should be routed via the tunnel. With this option, users will be able to continue using their regular applications and that traffic will still be router via the default gateway on the endpoint, but traffic headed for specific IP-addresses will be routed via the VPN tunnel.

In order to use this feature, we need to enable Split tunneling. This needs to be configured in the session policy under

Client Experience

- Split tunneling (Should be set to ON)

If we define this setting, we also need to define something called Intranet applications, where we define which IP segments that Gateway client should intercept and tunnel via the VPN client.

NOTE: We also have an option called Split-tunneling Reverse, this will in essence work as a direct opposite of regular setup. So if we define intranet applications and have split tunneling set to reverse, the NetScaler will not intercept traffic destined to the intranet applications, but will intercept all other traffic.

Next we need to configure Intranet applications, this can be done under Virtual Server → Intranet Applications → Intranet Applications and here under select intranet application we click on the + sign to add IP addresses or IP-ranges using subnet masks.

Important here that we choose interception type TRANSPARENT. Proxy type is only used for the older Java based plugin.

When defining an Intranet Application, we can define

- Protocol (Required)
- Destination type (Required, which can be either IP-address, IP-range or hostname)
- Address (Depending on what destination type we choose)
- Destination port (Not required, if left blank we have all ports)

After we have defined these settings on the virtual server and connect, again we will notice that only traffic bound to the intranet applications will be intercepted by the NetScaler Gateway VPN client.

Client IP pools

With VPN settings by default all clients will be sourced from the SNIP when connecting to the internal network after the client has successfully connected. Some specific applications might need that the end client has a unique IP address to connect with. In this case, we can configure IP pools for the VPN endpoints using Intranet IPs.

In order to configure Intranet IPs we go into the virtual server → Intranet IP → Intranet IPs(or IPv6)

In addition, from there we can either specify individual IP addresses or specify an IP range using subnet prefix.

VPN Virtual Server Intranet IP Binding	
<input type="button" value="Add Binding"/> <input type="button" value="Unbind"/>	
Intranet IP	Netmask
10.217.215.210	255.255.255.255
10.217.215.0	255.255.255.254
<input type="button" value="Close"/>	

Clientless Access

Clientless access allows us to setup a SSL VPN connection without the use of any agent or client installed, this is purely browser based connection.

From within the SSL VPN connection we can have SSO to backend resources, published bookmark for our users and even use it to access internal file shares or our outlook web access.

To activate Clientless Access, we need to set the virtual server in Smart Access mode

- ICA Only
- Enable Authentication
- Double Hop
- Down State Flush

NOTE: Clientless access requires Universal licenses

Then we need to define some settings in the session policy that is bound to the virtual server. Firstly, we need to active clientless access. We can also define URL for our web-based email service such as Outlook Web Access server.

URL for Web-Based Email

Split Tunnel*

Session Time-out (mins)

Client Idle Time-out (mins)

Clientless Access*

Clientless Access URL Encoding*

Clientless Access Persistent Cookie*

We can also define Access URL Encoding. This defines how the URL of the internal resource should look like for end-users.

Clear:

<https://nsgw.test.local/cvpn/https/ddc.test.local/>

Encrypted: (The URL is randomly generated each time a user clicks on the bookmark)

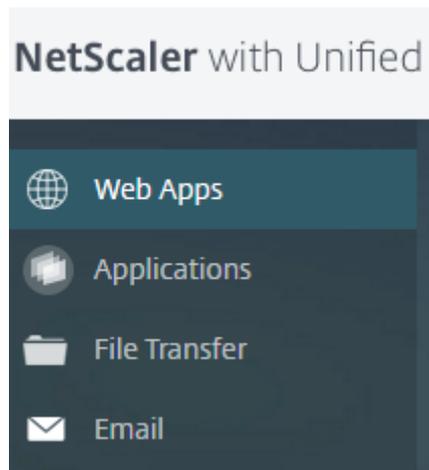
<https://nsgw.test.local/cvpn/M-iknskxUai2KqL2ycmzRKLlyheu4w/>

Obscure: (The URL is made unclear to the end-users but the URL remains the same each time, but the resource and the domain name is not revealed to the enduser)

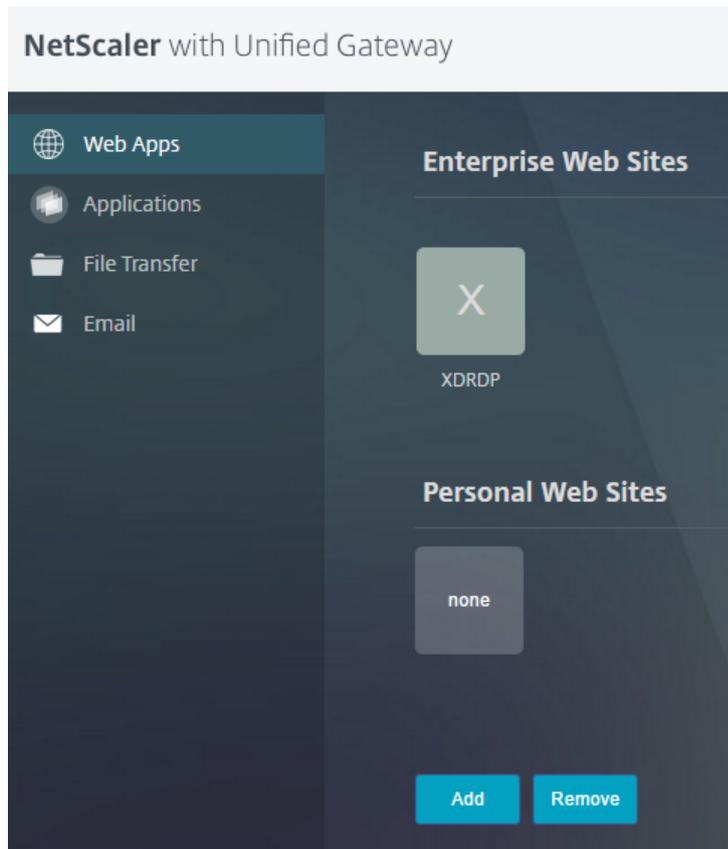


We can also configure Persistent cookie, which is used to do SSO to other backend resources such as SharePoint and so on.

NOTE: Defining an URL for web based email will give end-users a new Email icon available in the clientless access page when they login.



After these settings have been configured, and we login to our NetScaler Gateway virtual server we will be presented with this screen, of course depending on what resources that we already have bound to our users.



Adding resources

Now from within the clientless access screen, we have the option to pre add bookmarks to our users, which are either file shares or web URLs.

All users also have the ability to add their own web URLs which can be both internal and external URLs. If a user is adding URLs with an internal DNS prefix (Which has been specified under NetScaler Gateway → Global Settings → Clientless Access → Configure Domains for Clientless Access) the traffic will be routed via the NetScaler, if a user adds an external web address the browser will just open a new tab directly to that website without going through the NetScaler gateway.

For instance, under the clientless access pane as an end-user I can click on the Add button and I can from there enter the address of the resource I wish to add.

Add a Bookmark

To add a Web site, type in the full address, such as: `http://my.company.com/`.
To add a file share, type in the server and folder name, such as `\\fileserv\foldername`.

The maximum length of each field is 256 characters.

Name:

Address:

Description:

If I click Add now this will be added to my user and will always appear on my portal even if I logout and log back inn again. So all end user added bookmarks will appear under the pane **"Personal Web Sites"**



As an administrator I can also predefine resources I want all users to have. If I go into **NetScaler Gateway → Resources → Bookmark → Add**. From here I can customize a lot more settings on the resource I wish to deploy.

Create Bookmark

Name*

Text to display*

Bookmark*

Virtual Server

Icon URL ▼

Application Type

SSO Type

Use NetScaler Gateway As a Reverse Proxy

Comments

From here I can define an ICON, what kind of application type this is.

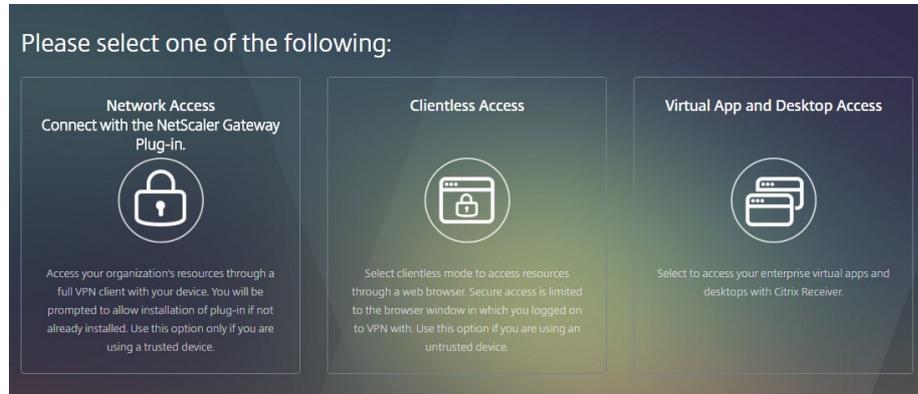
Binding the features together

Now we have gone through three different features such as ICA-Proxy, which the Citrix Receiver leverages, full VPN, which the Gateway endpoint client uses, and finally the use of clientless SSL VPN.

Now these features can live independently on its own virtual server or we can configure all features to be available from the same virtual server.

NOTE: Important note that if we put all these features on the same virtual server it would require it to run in smart access mode and therefore require Universal licenses regardless if users are only connecting using their Citrix Receiver

If we want to have a single web portal where we want to give the users the ability to choose the kind of resource they need, we need to make a change to the default session policy that they use. Under **Session Policy**, go to the request profile that is bound to it and then click on the **Client Experience** pane. Here, click on the **advanced** button. In the menu, we have an option called **Client Choices**. By enabling this, the users will get an option to choose what type of feature they need when logging in to the web portal, as shown in the following screenshot:



The options presented here are dependent on what is configured in the session policy. For example, if Clientless Access is not defined, it will not show up as an option here. If we have not entered a web interface address and STA is not available, Citrix XenApp will not show up as an option. Lastly, if we have set the virtual server to basic mode, it will automatically go to the StoreFront server after authentication.

Now we also have the option of enumerating the Citrix applications within the Clientless Access portal as well but it requires some extra configuration in order for it to work.

- Change some settings under `C:\inetpub\wwwroot\Citrix\storename` in the web.config file we need to change all three parameters that have the following settings:

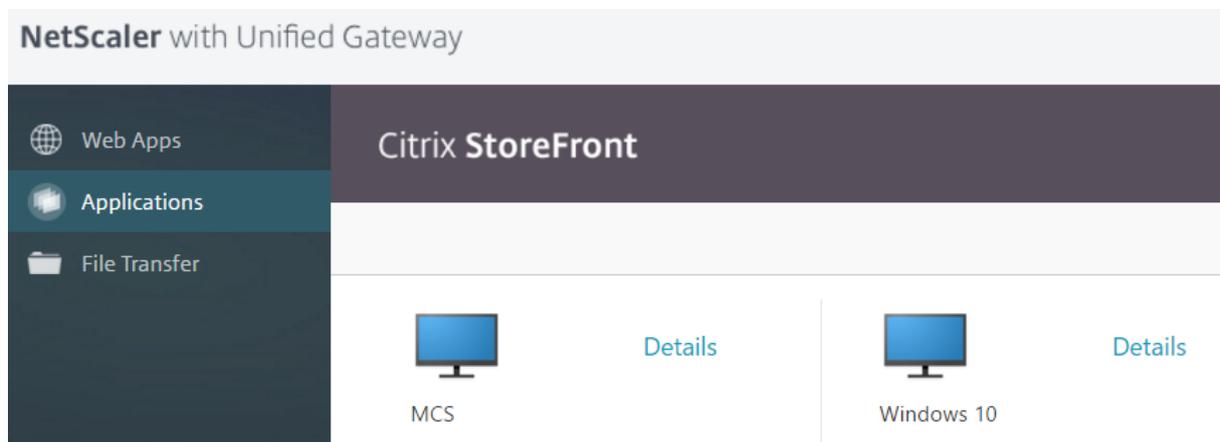
```
<add name="X-Frame-Options" value="deny" />
```

```
<add name="Content-Security-Policy" value="frame-ancestors 'none'" />
```

Change the values from "deny" to "allow" and from "none" to "self"

- Next, we need to add the AD domain that we are accessing, in order to allow Storefront integration to work with Clientless Access. This can be added under **NetScaler Gateway → Global Settings → Configure Domain for clientless access** followed by entering the domain name there.
- Lastly, we need to do an override global configuration and specify the NORMAL portal mode in the session policy under client experience, which is bound to the virtual server.

After these steps are done and a user logs in and chooses clientless access, the user will have the option to choose Applications from the menu as shown in the screenshot below.

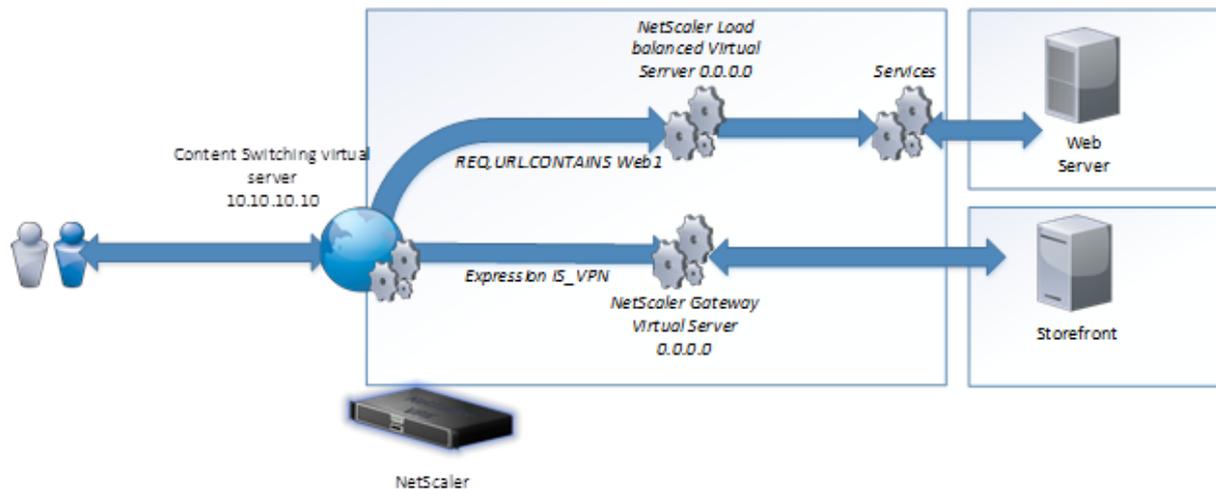


Unified Gateway

Unified Gateway is a new feature that was included in NetScaler 11, the idea behind it was to free up IP-addresses to be able to deliver multiple services behind a single IP-address. This is something that is achievable today using content switching which allows us to redirect traffic from a single IP to multiple load balanced virtual server using expressions. Problem was that this was never supported with NetScaler Gateway and therefore we needed to have one public IP for NetScaler Gateway and another IP for external load balanced services, but now with Unified Gateway this is now possible.

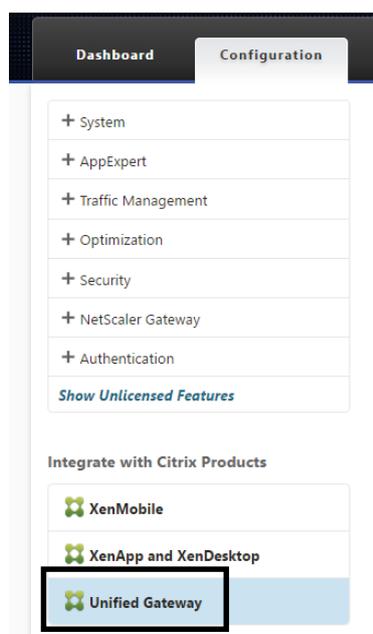
NOTE: Use of Unified Gateway requires either Enterprise or Platinum licenses, for a more detailed description of the licenses look at the NetScaler datasheet → https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/netscaler-data-sheet.pdf

So with this we can have one content switching virtual server which has an addressable IP address, which can then have one non-addressable NetScaler Gateway virtual server attached to it and then have one or multiple non-addressable load balanced virtual server attached to it, based upon expressions which are attached to the content switch.



So this example contains only one load balanced virtual server, if an end user goes to the IP 10.10.10.10 and has an URL, which contains web1 the content switch virtual server will see this and redirect this particular session to the backend load balanced virtual server which again is connected to a dedicated backend server. Now besides this there are no major differences between NetScaler Gateway and Unified Gateway.

To setup a Unified Gateway the simplest way is to run through the wizard in the UI. Now the wizard is pretty simple.



You define a virtual server name, IP and port, which will be the context of the content switching virtual server. Then define a server certificate and authentication point, and then

apply a portal theme to the NetScaler Gateway. Lastly we define applications which should be available in the which is the same way we assign URL resources or bookmarks when setting up regular NetScaler Gateway.

Unified Gateway Configuration

Virtual Server		
Virtual Server Name UF	IP Address 10.217.215.74	Port 443

Server Certificate ✎

! Certificate chain is incomplete. Upload a Certificate with the following subject : /DC=local/DC=test/CN=test-DC-CA ✕

 nsgw

Authentication	
Primary Authentication Active Directory/LDAP: AD	Secondary Authentication Not Configured

Portal Theme ✎

Applied Theme **X1**

Applications +

To add, please click on the + icon

After we are done with the wizard. We will have

- One addressable content switching virtual server
- One non-addressable NetScaler Gateway virtual server
- One policy on the content switching virtual server containing the expression `is_vpn_url` which is targeted to the NetScaler Gateway Virtual Server

Configure Content Switching Action

Name

Choose Virtual Server or Expression
 ▾

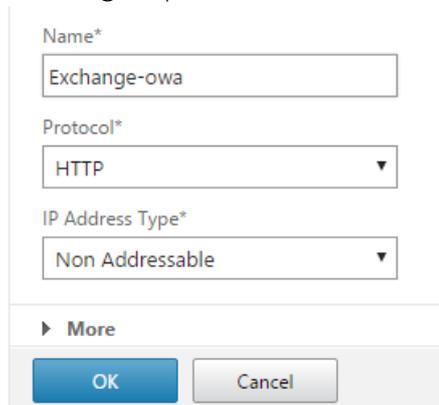
Target Virtual Server
 >

Comment

Now if we want to assign more resources to the Unified Gateway for instance other load balanced resources we can do this using Content Switching policies. For instance we can

setup a load balanced virtual server with a non-addressable ip 0.0.0.0 which points to an email-server backend.

- Setup a load balanced virtual server which points to a backend resource (service or service-group) use a non-addressable IP address.

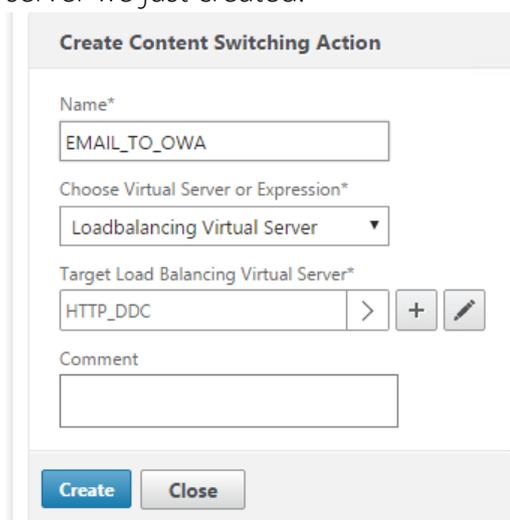


A screenshot of a configuration dialog box for a virtual server. It contains three dropdown menus: 'Name*' with the value 'Exchange-owa', 'Protocol*' with the value 'HTTP', and 'IP Address Type*' with the value 'Non Addressable'. Below these is a 'More' link with a right-pointing arrow. At the bottom are 'OK' and 'Cancel' buttons.

- After we have created the virtual server go into Content Switching → Actions → Click Add. Now there are multiple ways that we can define expressions here. In my example I have two URL which points to the same Unified Gateway external public IP.

mail.org & vpn.org If a user enters mail.org the content switching rules will point them to the load balanced virtual server we created. If a user enters vpn.org they will be pointed to the NetScaler Gateway virtual server which is attached to the content switching virtual server. Since the Gateway is already bound when we ran through the Unified Gateway wizard, we have to create the content switching policy for the mail server.

- First define the action, point it to a load balancing virtual server and define the server we just created.



A screenshot of the 'Create Content Switching Action' dialog box. It has a title bar 'Create Content Switching Action'. The form includes: 'Name*' with 'EMAIL_TO_OWA'; 'Choose Virtual Server or Expression*' with 'Loadbalancing Virtual Server'; 'Target Load Balancing Virtual Server*' with 'HTTP_DDC' and a right arrow, a plus sign, and an edit icon; and a 'Comment' text area. At the bottom are 'Create' and 'Close' buttons.

- Then go into Content Switching policies → And click Add. Then define an expression, the simplest is using HTTP hostname which contains mail and the

define the action we just created.

Create Content Switching Policy

Name*
EMAIL_OWA

Action
EMAIL_TO_OWA

Log Action

Domain

Expression URL

Expression*
Operators Saved Policy Expressions
HTTP.REQ.HOSTNAME.CONTAINS("mail")

Switch to Classic Syntax

Create Close

- Click Create. Then go into the Unified Gateway content switch virtual server. Then click Edit. Then under content switching policy binding click the pencil button to add another policy. Then click add binding. Then click select policy and add the policy we just created and click bind.

Policy Binding

Select Policy*
EMAIL_OWA

► More

Binding Details

Priority
600

Goto Expression
END

Invoke LabelType*
None

Target Load Balancing Virtual Server
Click to select

Bind Close

NOTE: We just defined the load balancing virtual server in the action, so do not define a load balancing virtual server under the target in this menu.

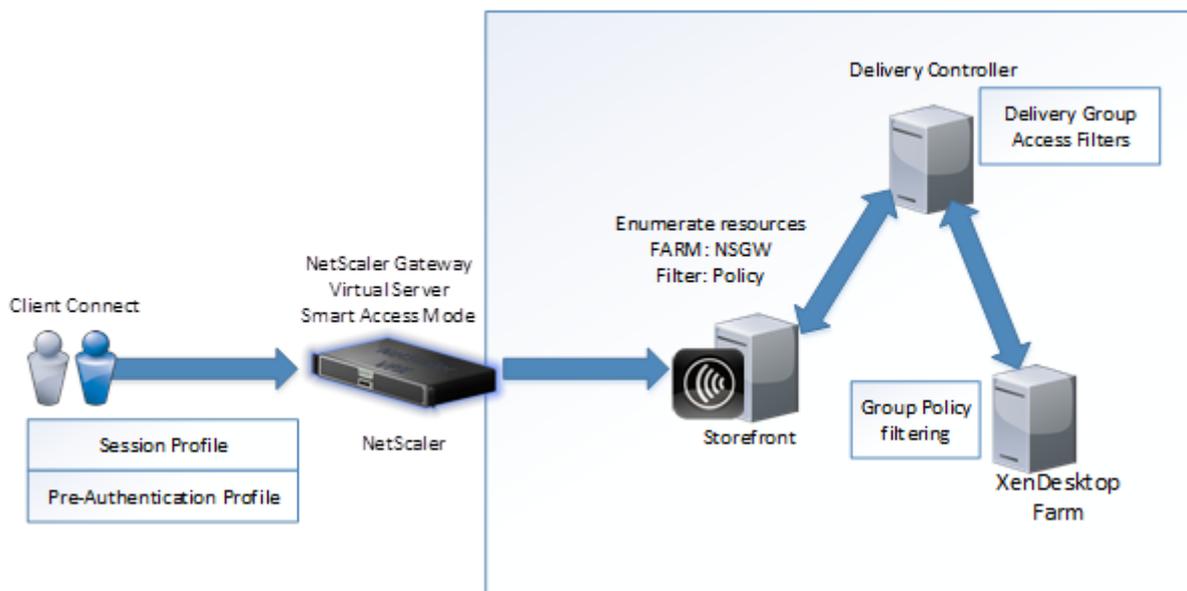
Then we should have two policies bound to the virtual server

Content Switching Virtual Server Content Switching Policy Binding			
Priority	Policy Name	Expression	Action
6000	EMAIL_OWA	HTTP.REQ.HOSTNAME.CONTAINS("mail")	EMAIL_TO_OWA
63000	UG_CSPOL_UF	is_vpn_url	UG_CSACT_UF

Smart Access – Access Policy

Smart Access is a feature, which combines policies from the NetScaler to acts as filters to then users connect to a Citrix environment. When a user connects and runs through a set of different policies, these policies will be used as filter. These filters will be sent to Storefront do use them to see if they are allowed/denied access to applications & desktops based upon those resources.

These filters can also be used for Group Policy processing. As an example we can create a policy which check if the end-user computer which is connect does have hard drive encryption activated before they are granted access to use drive mapping on a VDI session. We can also use regular session profiles based upon more general policies to grant access to applications or not.



There are two types of policies that can be used as filters. Either pre-authentication policies or session policies. These filters can then be used as access control for delivery

groups or group policy processing on and VDA agent. For instance, we can have a mix of different filters. This requires that we have Smart Access mode enabled on the virtual server, and that we setup XML trust on the XenDesktop farm we are using.

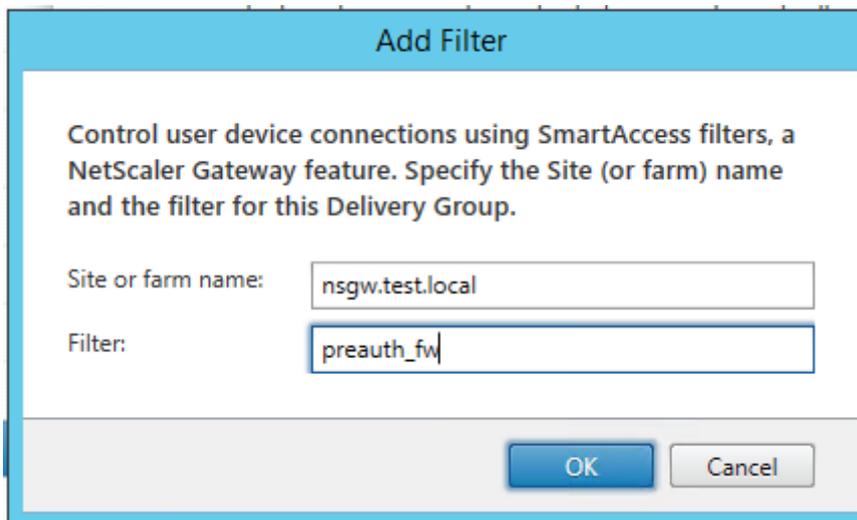
This XML trust can be configured from a Delivery Controller using PowerShell.

*asnp Citrix.**

set-brokersite -TrustRequestsSentToTheXmlServicePort \$true

After this is done, we can for instance bind a pre-authentication policy to a NetScaler Gateway virtual server, so as an example let's use that as an filter getting access to some specific desktop group.

We have a pre-authentication policy called **preauth_fw** bound to the NetScaler Gateway virtual server called **nsgw.test.local** so when going into Citrix Studio we need to use these values when defining access control. So for instance if we have a delivery group which some particular VDAs we want to limit access to, go into Studio → Delivery Groups → Right click on a delivery group and choose Edit Delivery Group → Go into Access Policy and click Add, specify the Virtual server name under site or Farm name and the policy name under filter.



Add Filter

Control user device connections using SmartAccess filters, a NetScaler Gateway feature. Specify the Site (or farm) name and the filter for this Delivery Group.

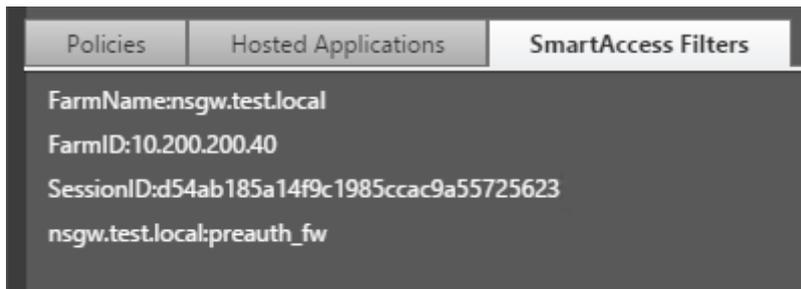
Site or farm name:

Filter:

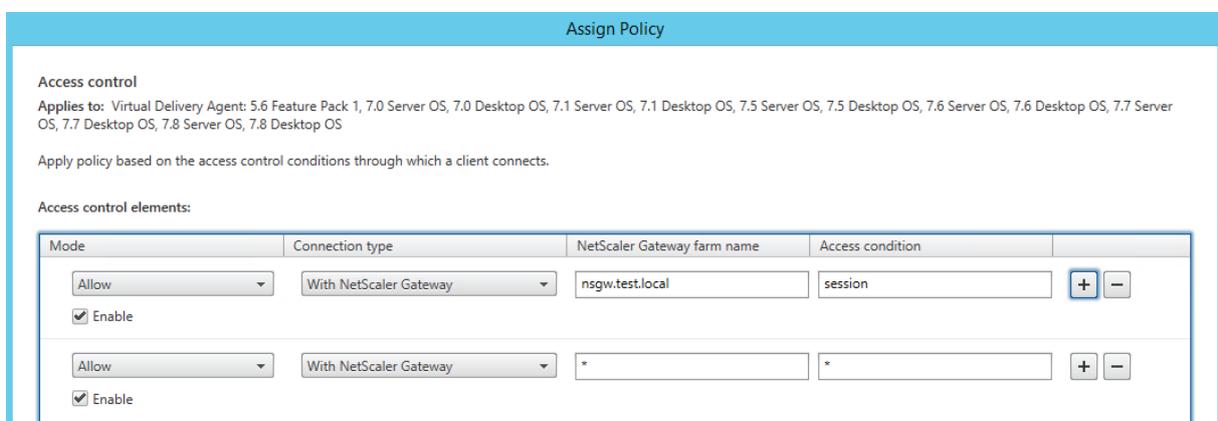
OK Cancel

Then click OK, then click OK and then close the delivery group menu. Now next time a user's logs inn you will notice that the user will only get access if the endpoint passes the preauthentication check.

This can also be viewed from within Citrix Studio after a user is logged inn and then going into session details and going into the smart access filters



These policies can also be used on Group Policy processing, for instance if we want to limit client drive mappings to certain amount of users based upon endpoint scans or network criteria, we can use the same filters on user policies. Go into policies, if we already have one created right click and choose edit. Choose users & machines and select under Access Control. From here, we can define the same filters again



To assign access based upon for instance session policies or even pre-authentication policies. In this example, I just used the name of my session profile called "session"

Smart Control – ICA Policies

Smart Control is a new feature within NetScaler Gateway which allows us to configure ICA policies for XenApp and XenDesktop from within NetScaler, instead of defining it within policies on XenDesktop/XenApp. Smart Control is implemented through ICA policies on NetScaler Gateway, each policy uses an expression and access profile that can be applied to users, group, virtual server or globally which are processed after a ICA session is established. To implement Smart Control, go into the GUI → NetScaler Gateway → Policies → ICA and click on Add. From here we use different expressions to filter based upon different attributes. Or we can use a general expression like "true" which for instance can be attached to a virtual server. Give the policy and name and click on the + sign under action. Under action we need to give it a name and then click on the + sign to create an ICA access profile.

So for instance we can create an ICA access profile which denies drive mapping access.

Create ICA Access Profile

Name*
Deny_drive_redirection

Connect Client LPT Ports
Default

Client Audio Redirection
Default

Local Remote Data Sharing
Default

Client Clipboard Redirection
Default

Client COM Port Redirection
Default

Client Drive Redirection
Disabled

Client Printer Redirection
Default

Multistream
Default

Client USB Drive Redirection
Default

After you are done defining the profile click create, click create on the action window.

Create ICA Action

Name*
deny_drive_action

ICA Access Profile*
Deny_drive_redirection

Create Close

And lastly click Create on the ICA policy.

Name*
all_users

Action*
deny_drive_action

Expression*
Operators Saved Policy Expressions Frequently Used Expressions
true

After we have created the policy, we need to bind it to an object, in this example we are going to use the NetScaler Gateway virtual server. Go into NetScaler Gateway → Virtual Server → Policies → click on the + sign and from the list choose ICA policy

Policies

Choose Policy*
ICA

Choose Type*
ICA Request

Continue Cancel

Click Continue and on the next window click on the "Click here to select policy"

The screenshot shows the 'Policy Binding' configuration window. It has two main sections: 'Policy Binding' and 'Binding Details'. In the 'Policy Binding' section, there is a 'Select Policy*' field with a 'Click to select' button, a right arrow, a plus sign, and an edit icon. In the 'Binding Details' section, there is a 'Priority*' field with the value '100' and a help icon, and a 'Goto Expression*' dropdown menu with 'END' selected.

And choose the pre created policy. Now that we have bound the policy to the virtual server

Group Based Access

Most profiles in NetScaler can be bound either to a virtual server, globally or to an AAA user or group. Most of the scenarios I have covered in this book has been focused on the virtual server level, therefore it is time to move into group based policy control.

It is however important to understand how policies apply on a NetScaler Gateway. As mentioned they can be applied in multiple levels, sorted by priority level where AAA users has the highest priority

- NetScaler Gateway globally
- AAA Globally
- NetScaler Gateway virtual server
- AAA Groups
- AAA Users

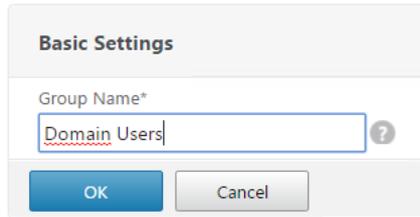
For instance, if we have a policy bound to a NetScaler Gateway virtual server and one to an AAA group, the settings defined in the policy for the AAA group will be applied. If we have settings defined in the NetScaler Gateway policy and not in the AAA group policy, the settings will be merged.

So important to remember, all policies will be default be merged unless specified in a policy which as higher priority.

Now if we want to define policies for certain AAA groups we first need to create the AAA group in NetScaler. Go into NetScaler Gateway → User Administration → AAA Groups, from there click Add

Now enter the name of the group we want to bind as policy to, as it is in Active Directory

AAA Group



Basic Settings

Group Name*

Domain Users ?

OK Cancel

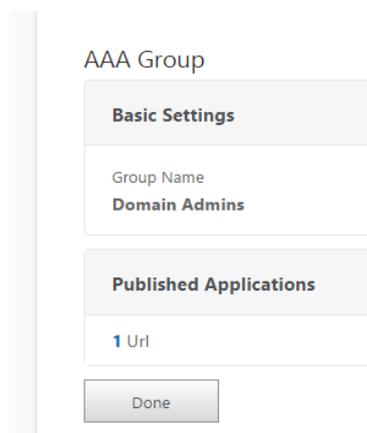
NOTE: The group name in AAA groups is case sensitive, so best practice is to just to a copy/paste of the group name

After we have created the group we will get the option to bind policies and resources to it. We can bind the following policies/resources to an AAA user or group

- Authorization Policies
- Session Policies
- Traffic Policies
- Audit Syslog
- Audit NSlog
- Published Applications
- Intranet IP Addresses
- Intranet Applications

NOTE: Authorization policies can only be bound to AAA users or groups not directly to a NetScaler Gateway virtual server

So as an example we can bind an RDP Bookmark to a particular AAA group. Open up the group under **NetScaler Gateway** → **User Administration** → **AAA groups** → **Edit**. Click **Published Applications** → Click **Add binding**, and select any preexisting bookmarks and click **bind**.



AAA Group

Basic Settings

Group Name

Domain Admins

Published Applications

1 Url

Done

After we are finished adding the resources for our AAA groups we can click Done and we will go back to the NetScaler Gateway previous menu.

Now when we mix multiple session policies based upon AAA users, groups and virtual servers it might be difficult to see which policies apply when a particular user logs on. It not that easy to see in real-time from the UI which policies that are applied. A simpler approach is to use the cli command

```
nsconmsg -d current -g pol_hits
```

This will in real-time list out all applied policies when a user logs on.

High availability

This section is going to focus on setting up a NetScaler in High-availability, to provide automatic failover properties for our remote access feature running on NetScaler.

NetScaler supports two types of high-availability features, one is active/passive and supports a maximum of two nodes (appliances) which is also known as high-availability in NetScaler. The other option is called clustering which uses Active/Active instances and supports up to 32-nodes.

The high-availability feature is available on both NetScaler Gateway appliances and on the NetScaler appliance, and is part of the platform license. Clustering however is only available on the regular NetScaler appliance and requires a particular license to be enabled.

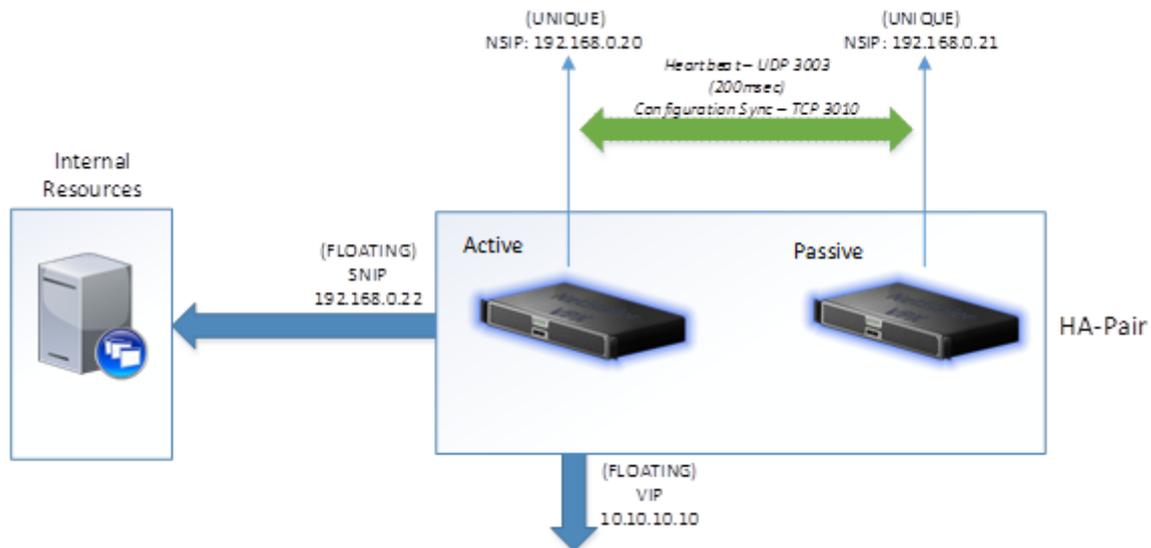
NOTE: This eBook does not cover the clustering feature, only high-availability.

High-availability gives us automatic failover in case of an appliance failure or NIC failure on a particular appliance. This typical scenario is having two appliances located on the same layer 2 subnet.

When setting up high-availability there are requirements for it to be supported

- Have the same firmware and running the same license
- Be of the same model version for instance (VPX 10 & VPX 10) not mixing different models and editions
- Both appliances have their unique NSIP, which they will use for exchanging configuration, and synchronize files.

The high-availability feature uses the NSIP on both nodes to send heartbeats across so see if the remote node is alive or not. This heartbeat feature is sent across UDP port 3003 each 200 milliseconds. It is also uses the terms Primary/Secondary, where the primary is the active node and the secondary is the passive node. If a node does responds to the heartbeat after 3 seconds it will be marked as down and the second node will resume control of the configuration and make itself, the primary node.



By default, the high-availability feature uses GARP to broadcast its MAC address via layer two

since both of the nodes has their own MAC address. When a device failover occurs, the secondary node sends out GARP packets to update the MAC table of nearby nodes (switches, routers, and firewalls) so that the new requests are sent to the new node. Also, NetScaler's in an HA pair will also share SNIP addresses, meaning that the passive node will have the IP address listed, but it will be listed as passive. Therefore, to set up NetScaler in an HA pair, we only need one IP for SNIP and two NSIP addresses.

It is important to note that some firewalls do not support GARP, or GARP is blocked, and therefore, we need to configure VMAC for the deployment. When using VMAC, the MAC address is shared between the two nodes, and therefore, it is not required to use GARP to update the MAC table on nearby nodes. I will come back to this later in the module and see how we can configure VMAC.

Open of the GUI on one of the nodes. Go into System → High-availability → Nodes → Click add. Under the Remote Node IP Address enter the NSIP of the other node, either using IPv4 or IPv6 address. Then enter the credentials for the remote appliance in the remote system login credentials.

All we need to do is enter the IP address, configure the remote system to participate, turn off HA monitors on interfaces that are down, and enter a different username and password if it differs from the node we are configuring it on.

Turning off HA monitors on interfaces that are down means that NetScaler will not try to send HA probes from one node to another on interfaces that are not in use.

The last option is that INC is needed if the appliances are on different subnets and therefore require independent network configurations, since the default option sets them up using the same network layer two network.

After we have entered the information and clicked on **OK**, the primary node will start to propagate its information and configuration with the secondary node and set up a high availability pair.

It will also start to synchronize files such as SSL certificates and application firewall XML files; you can view the different files that are part of the synchronization process at <http://support.citrix.com/article/CTX138748>.

It is important to note that there are a few items that will not be synchronized, and these are licenses and rc.conf files. Irrespective of whether the HA pair is active/passive, you will still need one active license of each of the nodes.

NOTE: By using the command in CLI *show ha node* we can easily see which nodes in a HA-pair that are primary and secondary, we can also view this from the high-availability

screen within the GUI, and we can always from the GUI see which node we are logged into, in the top right corner



High-availability nodes can have multiple statuses.

- ENABLED
- STAY PRIMARY
- STAY SECONDARY
- DISABLED

The status of the appliances defines how the work in case of a triggered failover or in case of unplanned outage. By default, after setting up, high-availability both appliances are set in ENABLED mode where one is primary and the other one is secondary

The table below shows the different statuses and effect when something happens.

Node1	Node2	Force Failover	Unplanned Failover
Primary	Secondary	Node1 and Node2 will switch status. Node2 becomes Primary	Node2 will take Primary role
STAY PRIMARY	Secondary	Will not work, will get an error message	If Node1 goes down Node2 will resume control until Node1 returns back to normal state
STAY PRIMARY	STAY SECONDARY	Will not work, will get an error message	HA-pair goes down, Node2 will not resume control of the HA-pair. Normal state returns with Node1 comes back online
STAY SECONDARY	Primary	Will not work will get an error message	HA-pair goes down if Node2 goes offline. Normal state returns when Node2 comes back online

From the GUI we have options to do both force failover and force

synchronization. We also have an option to edit each node directly but only the one that we are directly logged onto.

Cross-subnet High-availability

If we want to achieve high availability across different subnets, we need to alter some configurations on the HA pair. For instance, if we have another NetScaler appliance in a different subnet, it might require a different route table or even other NAT rules in place in order to communicate with the backend resources. In a traditional HA setup, all the different static/dynamic routes are synchronized across the HA pair, and they also share the same SNIP address.

In order to make this work, we need to configure **Independent Network Configuration** under the **HA** settings. This can be done using the command:

Add ha node id IPAddress -inc ENABLED

Using this command, the NetScaler, by default, does not propagate settings that are unique to that particular network, such as:

- IPs (NSIP, MIP, and SNIP)
- VLANs
- Routes
- Dynamic Routing
- Reverse NAT

Therefore, these settings need to be configured individually on each of the two nodes. Other settings are still propagated in the same matter as before.

This allows us to have a unique configuration for each node in the HA pair. It is important to note that the VIP is floating between the HA-pair; therefore, even if the nodes are located across different subnets, the NetScaler's still need to be connected to a layer 2 network for the VIPs to share the same network and be able to failover if needed.

Failsafe mode

In a high availability configuration, fail-safe mode ensures that one node is always primary when both nodes fail the health check. Fail-safe mode ensures that when a node is only partially available, backup methods can activate and can handle traffic. You configure high availability fail-safe mode independently on each node.

The table below shows some of the fail-safe cases.

Node A (Primary)	Node B (Secondary)	Default HA behavior	Failsafe HA behavior	Description
---------------------	-----------------------	------------------------	-------------------------	-------------

HA failed, last	HA failed, first	(A)Secondary, (B)Secondary	(A)Primary, (B) Secondary	If both nodes fail, one after the other, the node that was the last primary remains primary.
HA failed, first	HA failed, last	(A)Secondary, (B)Secondary	(A)Secondary, (B)Primary	If both nodes fail, one after the other, the node that was the last primary remains primary.
HA failed	UP (StaySecondary)	(A)Secondary, (B)Secondary	(A) Primary, (B) Secondary	If the secondary is configured as STAYSECONDARY, the primary remains primary even if it fails.

Go into the UI → System and then click High Availability. In the details pane, on the Nodes tab, select a node and then click Edit. In the Configure Node dialog box, under Fail-Safe Mode, select Maintain One Primary node even when both nodes are unhealthy and then click OK.

The screenshot shows the 'Configure Node' dialog box with the following fields and options:

- ID: 0
- IP Address: 10 . 217 . 215 . 108
- High Availability Status*: ENABLED (Actively Participate in t ▼)
- HA Synchronization**
 - Secondary node will fetch the configuration from Primary
- HA Propagation**
 - Primary node will propagate configuration to the Secondary ?
- Fail-safe Mode** (highlighted with a red box)
 - Maintain one primary node even when both nodes are unhealthy

VMAC

In a high availability setup, the primary appliance owns all the floating IPs (SNIP, VIP) This appliance is the one that responds on all the ARP requests for these addresses that occur

on the network. In case of a failover, the secondary appliance will broadcast a (Gratuitous ARP)GARP update to the network to update the MAC tables to point to the other appliance.

Some devices do not support GARP, therefore after a failover occurs we might get downtime. Now NetScaler has something called VMAC to overcome all these GARP issues. When you configure a VMAC on NetScaler appliances, both appliances participating in high availability setup possess identical MAC addresses. Therefore, when failover occurs, the MAC address of the secondary appliance remains unchanged, and ARP tables on the external devices need not be updated.

VMAC is a user-defined MAC address that is shared by the primary and secondary appliances in the high availability setup.

VMAC can be created in the UI under System → Network → VMAC. From there specify a RouterID, this is used for generating of the MAC address. The generic MAC address that is used is in the form of 00:00:5e:00:01: and the RouterID is a number that defines what the last octet is going to be. So for instance if we specify a router ID of 3 the mac address is going to be

00:00:5e:00:01:03

Then lastly specify which interface this should be on, this of course needs to be the interface where the VIPs are serviced

Virtual Router ID*

Priority

Tracking*

Preemption

Preemption Delay Timer (secs)

Reduced Priority

Sharing

Interfaces bound to this vrid Interfaces tracked for this vrid

Associate Interfaces

Available (1) Select All

1/1 +

Configured (1) Remove All

0/1 -

Then click Create. The settings will propagate on the secondary appliance as well and they will not share the common mac address.

Upgrading

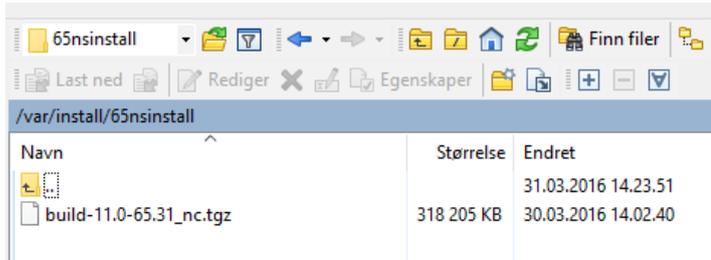
When doing firmware upgrades of a NetScaler HA-pair it should be done, in a specific sequence to ensure no downtime of the appliances and normal operations. Before an eventual upgrade make sure that both NetScalers support the new firmware and make sure to properly read the release notes to make sure that there are now new default changes to the security like with SSL/TLS and such.

Most release notes can be located on the same page where we download firmware upgrades from mycitrix.com or at docs.citrix.com → <http://docs.citrix.com/en-us/netscaler-gateway/11/whats-new.html>

When it is time to upgrade, we must always start with the secondary node.

- Log into the secondary appliance using SSH (You will get notified that you are logging into the secondary node)
- Run the **save config** command to ensure that configuration is properly saved.
- After config is saved log into shell by typing **shell**

- From there change the working directory to `/var/install` by using the command `cd /var/install`
- From there create a new folder for the new firmware. A typical way is to use the firmware version name. For instance, `mkdir 65nsinstall`
- Now connect to the same NetScaler appliance using a FTP/SFTP client for instance using FileZilla or using WinSCP and copy the download firmware tgz file to that particular folder we created under `/var/install/65install`



- Switch back to the CLI and extract the file in the folder using the command `tar -zxvf file.tgz`

If you are unfamiliar with the attributes, the z (is for file type gzipped) the x (to extract) the v (Print file names as they are extracted) -f (Use the following tar file for operation) So this command will extract the tar file within the same folder.

- Run the command `./installns` This will then trigger the installation process from within the same folder

```

root@nsvpx2# ./installns
installns: [1557]: BEGIN TIME 1459428611 Thu Mar 31 12:50:11 2016
installns: [1557]: VERSION ns-11.0-65.31.gz
installns: [1557]: VARIANT v
installns: [1557]: No options

installns version (11.0-65.31) kernel (ns-11.0-65.31.gz)

installns: [1557]: installns version (11.0-65.31) kernel (ns-11.0-65.31.gz)

The Netscaler version 11.0-65.31 checksum file is located on
http://www.mycitrix.com under Support > Downloads > Citrix NetScaler.
Select the Release 11.0-65.31 link and expand the "Show Documentation" link
to view the SHA2 checksum file for build 11.0-65.31.

There may be a pause of up to 3 minutes while data is written to the flash.
Do not interrupt the installation process once it has begun.

Installation will proceed in 5 seconds, CTRL-C to abort

```

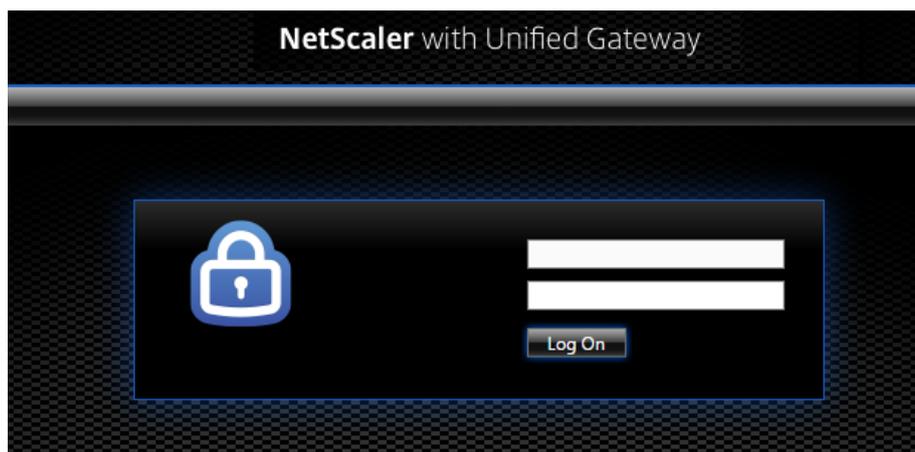
- When the installation process is complete, you will be given the option to reboot the appliance. Click on Y and press Enter.
- You can also notice that after the firmware upgrade process the appliances will turn off auto synchronization.

Master State	Node State	INC	Synchronization State
Primary	● Up	DISABLED	AUTO DISABLED
Secondary	● Up	DISABLED	AUTO DISABLED

- Next we need to switch the upgraded appliance to primary, this can be done using the CLI command **force failover** or by triggering it from the high-availability menu option. Note that you will be notified of HA version mismatch but this is expected.
- After the failover is successful, repeat the installation process on the other node.
- After the upgrade is complete on the second node in the HA-pair and the HA-version are the same, we can notice that synchronization will be enabled automatically again.

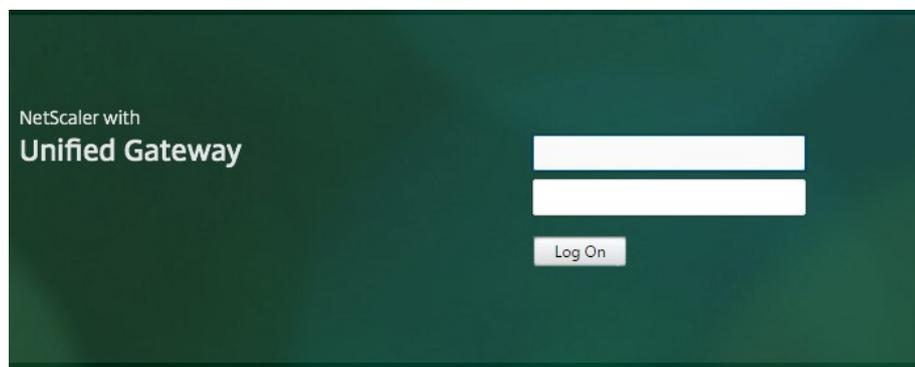
Portal customization

Starting with NetScaler v11 it is now a lot easier to do portal customization. By default, when we set up a NetScaler Gateway is still uses the old default portal theme.



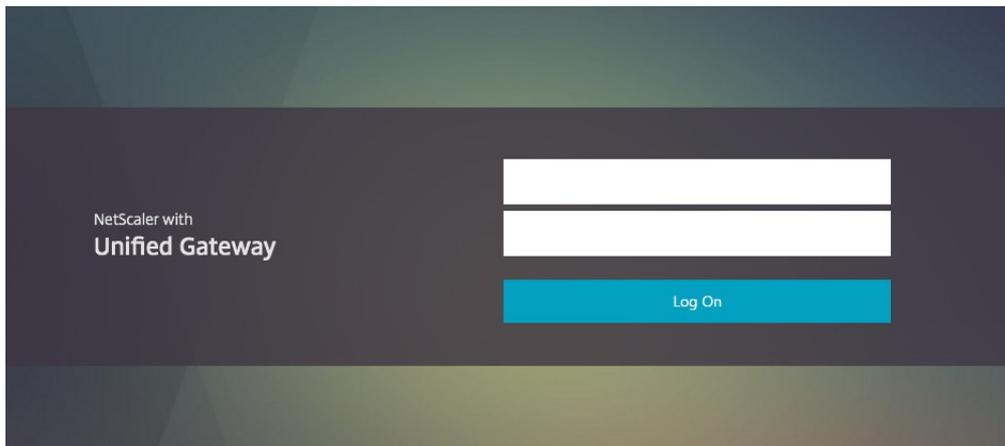
Default-theme (bind vpn vserver nsgw -portaltheme default)

We also have other portal themes includes as part of the gateway feature as well such as "Green Bubble"



Green Bubble (bind vpn vserver nsgw -portaltheme greenbubble)

And "X1" which gives a consistent look at feel which also compares with the newer Storefront UI



X1-theme (bind vpn vserver nsgw -portaltheme X1)

These themes can also be modified using the GUI, by going into **NetScaler Gateway** → **Virtual Server** → **Edit the particular virtual server** → **Portal themes** and changing the value from there. You also now have the option to do theme customization directly from within the NetScaler UI as well. This can be done by creating a custom theme based upon one of the three existing templates.

Go into **NetScaler Gateway** → **Portal Themes** → **Click Add** → Enter a custom name, choose an existing template, and click OK. We will now come into the portal theme customizer menu, from here; we can customize a lot of different settings which we can define such as color, menu options, logos and so on.

After we now have modified our theme and clicked OK, we are given the option to specify a particular language, by default this is set to English.

After we click OK here, we are given the option to alter for instance error messages or notification messages which appear in the portal.

Login Page

The Login Page is the first page presented to a VPN user. The Login Page is where the user enters their authentication information.

Page Title NetScaler Gateway	User Name Field Title User name
Form Title Please log on	Password Field Title Password
	Password Field2 Title Password 2

NOTE: The previous option that we had in earlier versions of NetScaler, which allowed us to customize directly css files are no longer supported by Citrix. The only options that we have are the ones that are directly available from within the UI. There are however some workarounds that work which can be found here →

<http://discussions.citrix.com/topic/367268-netscaler-11-custom-theme/>

We can also now configure an EULA to show on the end portal before users are allowed to connect.

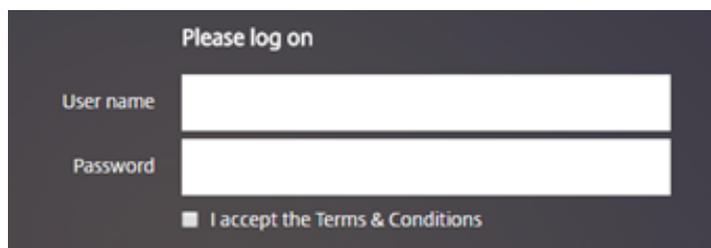
Binding an EULA to the Portal

With NetScaler there is now an option to bind an EULA (End User License Agreement) which will be shown to the users which they are required to accept before they are allowed to login.

NetScaler Gateway → Resources → EULA. From here we can create different EULA for different virtual servers. After we have created an EULA we have to bind it to the particular virtual server we want it displayed.

NetScaler Gateway → Virtual Server → EULA → Add binding, and choose the EULA we just created.

When users now try to login they will see this in the portal.



The image shows a dark-themed login portal. At the top, it says "Please log on". Below this, there are two input fields: "User name" and "Password". At the bottom of the form, there is a checkbox labeled "I accept the Terms & Conditions".

Security settings

When setting up a NetScaler Gateway it will be in most cases open externally for remote access to deliver Citrix to remote workers. Now by exposing a service externally you also open up yourself for attacks. There are many possible attack vectors

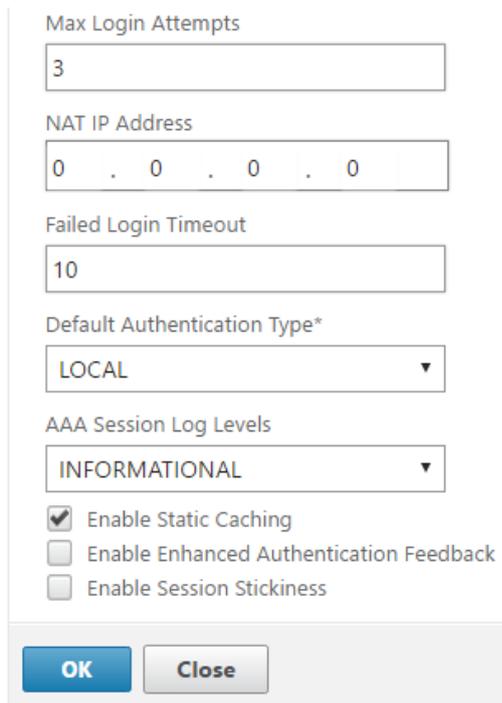
- Bruteforce attacks
- DDoS
- Protocol weakness
- Security exploits

Therefore, it is important to think about this when setting up NetScaler Gateway virtual server. Now when setting up a smart access server and allowing full VPN access for your endpoints you need to take extra care when setting up our policies. Therefore, this section

is separated into different groups which list different settings we can configure to have a higher level of security on our virtual server.

General settings:

Under NetScaler Gateway → Global Settings → Change authentication AAA settings → Define **Max Login Attempts** and then define **Failed Login Timeout**. This help to avoid dictionary attacks by locking out authentication attempts after a certain amount of attempts.



The screenshot shows a configuration dialog box with the following fields and options:

- Max Login Attempts:** A text input field containing the number 3.
- NAT IP Address:** A text input field containing 0 . 0 . 0 . 0.
- Failed Login Timeout:** A text input field containing the number 10.
- Default Authentication Type*:** A dropdown menu with 'LOCAL' selected.
- AAA Session Log Levels:** A dropdown menu with 'INFORMATIONAL' selected.
- Enable Static Caching**
- Enable Enhanced Authentication Feedback**
- Enable Session Stickiness**

At the bottom of the dialog are two buttons: 'OK' and 'Close'.

Here we also have the **enhanced authentication feedback** button, which helps end users by notifying them what is wrong when they try to login, but it can also expose some critical information to malicious attackers.

This setting can either be defined globally or per virtual server, but if we are using multiple virtual servers the best is to configure this globally so it affects all virtual servers.

Session Policies:

If we are implementing a full VPN solution, we can also specify multiple settings depending on what we want. The best practice is to not specify full access but use Split tunneling and specify intranet applications for those applications that the end-users need access to. This way only traffic destined to those applications will be processed by the NetScaler Gateway plugin.

In most cases also an end-user might not require access for a really long period of time and might forget to disconnect the session. In that case we can setup a timeout which decides when a session should be forcefully disconnected. This is done under **session policies** → Network Configuration → Advanced Settings.

Forced Timeout

Forced Time-out Warning (mins)

It is also useful to have more specific session policies depending on what type of resource is trying to connect. For instance, we can have a session policy using OPSWAT expressions to avoid non-healthy endpoints to connecting to our environment.

For instance, a session policy with OPSWAT rules to determine if the endpoint is running an authentic antivirus solution

Name

Profile*

Expression*

Operators	Saved Policy Expressions	Frequently Used Expressions
<code>CLIENT.APPLICATION(ANTIVIR_0_AUTHENTIC_==_TRUE_RTP_==_TRUE[COMMENT: Generic Antivirus Product Scan]) EXISTS</code>		

If the endpoint does not match the requirements, they will not get any access to the Citrix environment. The problem with this is that it happens after authentication has occurred, we can also use Preauthentication policies to do health checks before authentication, but then we cannot filter based upon AAA groups and users for instance.

In addition, we can use these settings in conjunction with Smart Access to control how the access to the Citrix environment and which group policies should be processed.

We can also specify idle-time out values, in the session profile together with split tunneling and session time-out

Split Tunnel*

Session Time-out (mins)

Client Idle Time-out (mins)

Now again an issue is if an attack has access to an end-users username and password and even has access to the end users device, then the attack will be able to access the environment. When possible try to add a two-factor authentication feature to minimize these types of attacks.

That way even if an attacker has access to the end users username and password they will not be able to login to the environment.

In addition, if we are not using Split tunneling, we should configure Authorization rules, which we can bind to the NetScaler gateway to define, ALLOW/DENY rules to internal resources using client expressions, which are then bound to AAA users or groups.

If this is not possible. Define ACL rules based upon the Intranet IP range that is defined as part of the NetScaler Gateway.

Now a lot of people focus on the SSL/TLS configuration of the virtual server, while that itself is important it should be part of the bigger picture since that only addresses the protocol exploits of SSL/TLS and might allow a malicious attacks to decrypt the secure connection and then do MiTM, while theoretically possible not easily achieved.

Now by default when configuring SSL/TLS Settings on NetScaler we can either use SSL profiles or use SSL parameters for each virtual server. If we use profiles, we cannot configure SSL parameters and the other way around.

NOTE: We also have the option to enable a global default SSL profile, which will be attached to all SSL protocol based virtual servers. This will use the ns_default_ssl_profile_frontend policy for front-end facing virtual servers. This can be enabled under Traffic Management → SSL → Change advanced SSL settings → enable default SSL profile, and take note after you enable it you cannot disable it.

The different SSL profiles can be viewed under System → Profiles → SSL Profile, by default there are two profiles one for front-end connections (for instance virtual servers) while the other are for backend connections (services, service groups)

Now there four main features that effect the security using TLS/SSL protocol

- Certificate (Private Key size, what does the certificate support?)
- Protocol Use (SSL or TLS?)
- Ciphers (Define how strong algorithm that should be used for encryption and which algorithm should be used for authenticity and authentication) Ciphers are attached to an SSL profile as well.

NOTE: There is a website called ssllabs.com, which is commonly used in conjunction with testing SSL/TLS security level on web services, where the score goes from F to A+ where A+ is the best possible score. This can only be achieved on the Gateway virtual server if it

only uses the more secure protocols and Ciphers which give a high level of encryption and if we have a valid certificate. Again, I have to emphasize that this only addresses protocol weaknesses.

For our virtual server to score A+ on ssllabs.com test there are some modifications that need to be done again the SSL Profile or using SSL parameters.

- Bind the entire certificate chain to the virtual server, which means the certificate, any intermediate certificates and root certificates
- Deny SSL Renegotiation (This is used from a client to renegotiate which protocol to use, which might be used for attackers to lower a session from TLS 1.2 to a SSL version with lower security. Settings it to FRONTEND_CLIENTSERVER will disallow renegotiation.

Deny SSL Renegotiation*

FRONTEND_CLIENTSERVER ▼

- Make sure that SSL3 is disabled (This is disabled by default in the default profiles and should be reflected in the frontend profile)

Protocol	
<input type="checkbox"/>	SSLv3
<input checked="" type="checkbox"/>	TLSv1
<input checked="" type="checkbox"/>	TLSv11
<input checked="" type="checkbox"/>	TLSv12

- Specify a supported Cipher group, which ensures a high-level of encryption, this is added under the SSL profile as well. A Cipher group specified which SSL/TLS protocol that should be used and which type of encryption.

Another thing to be aware of is that some options are available for only front-end connections, but not backend connections. Another thing is that not all ciphers are available for VPX editions. If you try to create an cipher group of ciphers which are not supported on the VPX you will get an error message.

- The simplest way is to create a cipher group using CLI:

VPX Example:

```
add ssl cipher vpx-ciphers
```

```
bind ssl cipher vpx-ciphers -cipherName TLS1.2-ECDHE-RSA-AES-128-SHA256
```

```
bind ssl cipher vpx-ciphers -cipherName TLS1-ECDHE-RSA-AES256-SHA
```

```
bind ssl cipher vpx-ciphers -cipherName TLS1-ECDHE-RSA-AES128-SHA
```

```
bind ssl cipher vpx-cipher-list -cipherName TLS1-AES-256-CBC-SHA
```

```
bind ssl cipher vpx-cipher-list -cipherName TLS1-AES-128-CBC-SHA
```

- **MPX Example:**

```
add ssl cipher mpx-ciphers
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-ECDHE-RSA-AES256-GCM-SHA384
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-ECDHE-RSA-AES128-GCM-SHA256
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-ECDHE-RSA-AES-256-SHA384
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-ECDHE-RSA-AES-128-SHA256
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1-ECDHE-RSA-AES256-SHA
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1-ECDHE-RSA-AES128-SHA
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-DHE-RSA-AES256-GCM-SHA384
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1.2-DHE-RSA-AES128-GCM-SHA256
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1-DHE-RSA-AES-256-CBC-SHA
```

```
bind ssl cipher mpx-ciphers -cipherName TLS1-DHE-RSA-AES-128-CBC-SHA
```

- Implement HSTS and HTTP -> HTTPS redirection

One of the last things we need to configure is HSTS (HTTP Strict Transport Security) which is a security mechanism which is in place to protect websites against protocol downgrade attacks and cookie hijacking. It allows the NetScaler to notify web browsers that it should only interact with its services using HTTPS. This is a feature which Google implemented into Chrome, but other browsers such as Firefox and Internet Explorer now support it. In order to configure it there are multiple steps.

- Have a valid certificate on the web-service (Root, any intermediate and server CA)
- Redirect all traffic from HTTP to HTTPS

- Serve an HSTS header on the base domain for HTTPS requests with header
`Strict-Transport-Security: max-age=10886400; includeSubDomains; preload`
- After this is done we can submit this to the Google chrome preload list here →
<https://hstspreload.appspot.com/>

Now to first do implement HTTP to HTTPS the simplest way is to setup a simple load balancing virtual server on HTTP port 80 using the same IP as the NetScaler Gateway virtual server and then setting up a redirect.

NOTE: If you use the NetScaler Gateway wizard in NetScaler to configure NetScaler Gateway it uses this setup to configure HTTP to HTTPS redirect.

Go into Traffic Management → Load balancing → Virtual Servers. Click Add and give it a descriptive name and enter the same IP address of the NetScaler Gateway virtual server, and using HTTP as protocol as port 80.

The screenshot shows a configuration dialog box for a virtual server. The fields are as follows:

- Name***: HTTP_HTTPSREDIRECT
- Protocol***: HTTP (selected from a dropdown)
- IP Address Type***: IP Address (selected from a dropdown)
- IP Address***: 10 . 200 . 200 . 40 (with an IPv6 checkbox to the right)
- Port***: 80

At the bottom, there is a "More" section with "OK" and "Cancel" buttons.

Click OK, when asked to bind service to the virtual server click Continue. Click on the protection pane on the right side and there under Redirect URL, ether the FQDN of the NetScaler Gateway virtual server using HTTPS.

The screenshot shows the "Protection" configuration pane for a virtual server. The fields are as follows:

- Redirect URL**: https://nsgw.msandbu.org
- Backup Virtual Server**: (empty dropdown menu)
- Disable Primary When Down**: (checkbox, currently unchecked)

After that click OK and we are done.

Then we need to implement and HTTP rewrite policy that can insert the HSTS header. Go into AppExpert → Rewrite → Go into Actions first and click Add.

Give it a name like INSERT_HSTS_HEADER, under type choose INSERT_HTTP_HEADER, under header name enter Strict-Transport-Security under expression enter "max-age=157680000" and then click Create.

Create Rewrite Action

Name*
insert_sts_header

Type*
INSERT_HTTP_HEADER

Use this action type to insert a header.

Header Name*
Strict-Transport-Security

Expression

Operators Saved Policy Expressions Frequently Used Expressions

"max-age=157680000"

Then go back to the rewrite menu. Go into Policies and then click Add. Give it a name IMPLEMENT_HSTS_HEADER for instance and under Action choose the rewrite action we created, under expression use the expression true

Create Rewrite Policy

Name*
IMPLEMENT_HSTS_HEADER

Action*
insert_sts_header

Log Action
-

Undefined-Result Action*
-Global-undefined-result-action-

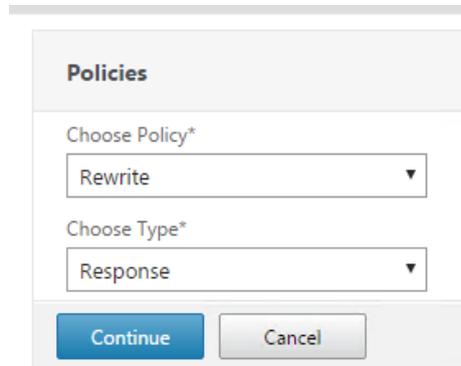
Expression*

Operators Saved Policy Expressions Frequently Used Expressions

true

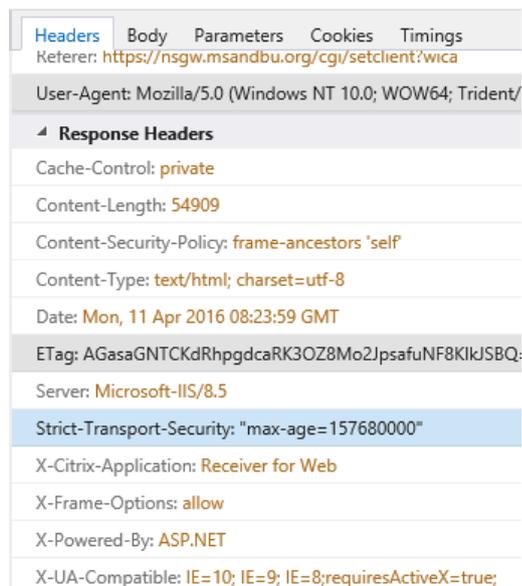
Comments

Then click add. After we are done with this we need to add the rewrite policy to the NetScaler Gateway virtual server. To into NetScaler Gateway → Virtual Server → Choose the existing virtual server click edit → Policies, choose Rewrite and choose Response.

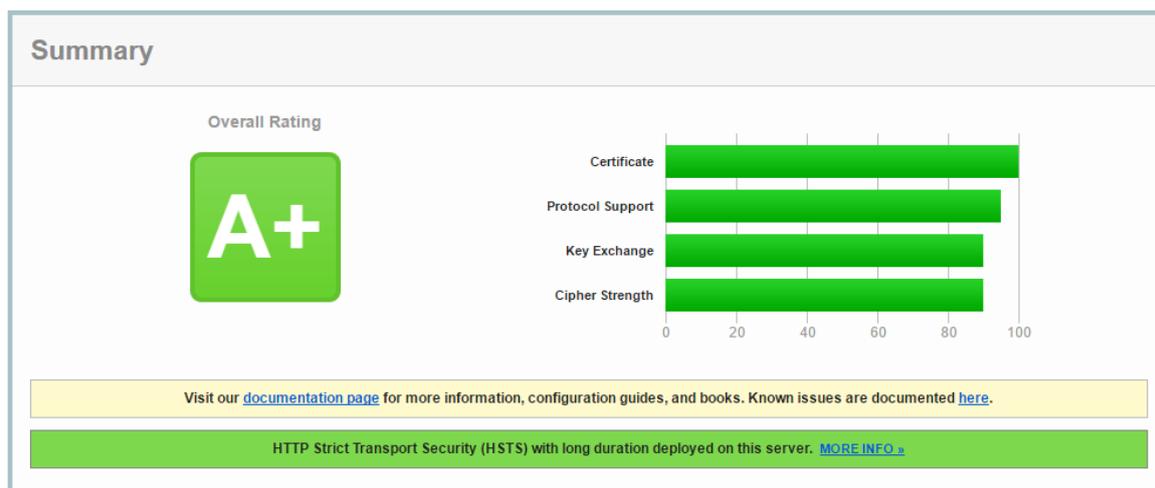


And then bind the existing rewrite rule we created, and click OK, and then we are done with the HSTS configuration.

The simplest way to confirm the HSTS settings and ciphers are properly setup is either using SSLlabs.com and do a test or using developer tools in Internet Explorer. This can be accessed clicking F12 within Internet Explorer and look at the HTTP header when connecting to the NetScaler Gateway virtual server.



From SSLlabs.com



NOTE: The simplest way to test ciphers groups when configuring NetScaler Gateway is using OpenSSL, which can be used for this purpose more info on this blogpost here → <http://bit.ly/1MPaGY6>

Authentication and Authorization

SAML Authentication

Now by default, Citrix XenDesktop 7.8 does not support SAML based authentication. That solution is as of now only supported for XenApp 6.5. This section is based upon a tech preview which Citrix has available now which delivers this feature for XenDesktop 7.8.

It should be noted however that this feature only works for Receiver for Web and not the native Receiver client. The knowledge article and more information on how to download the different components can be found here -->

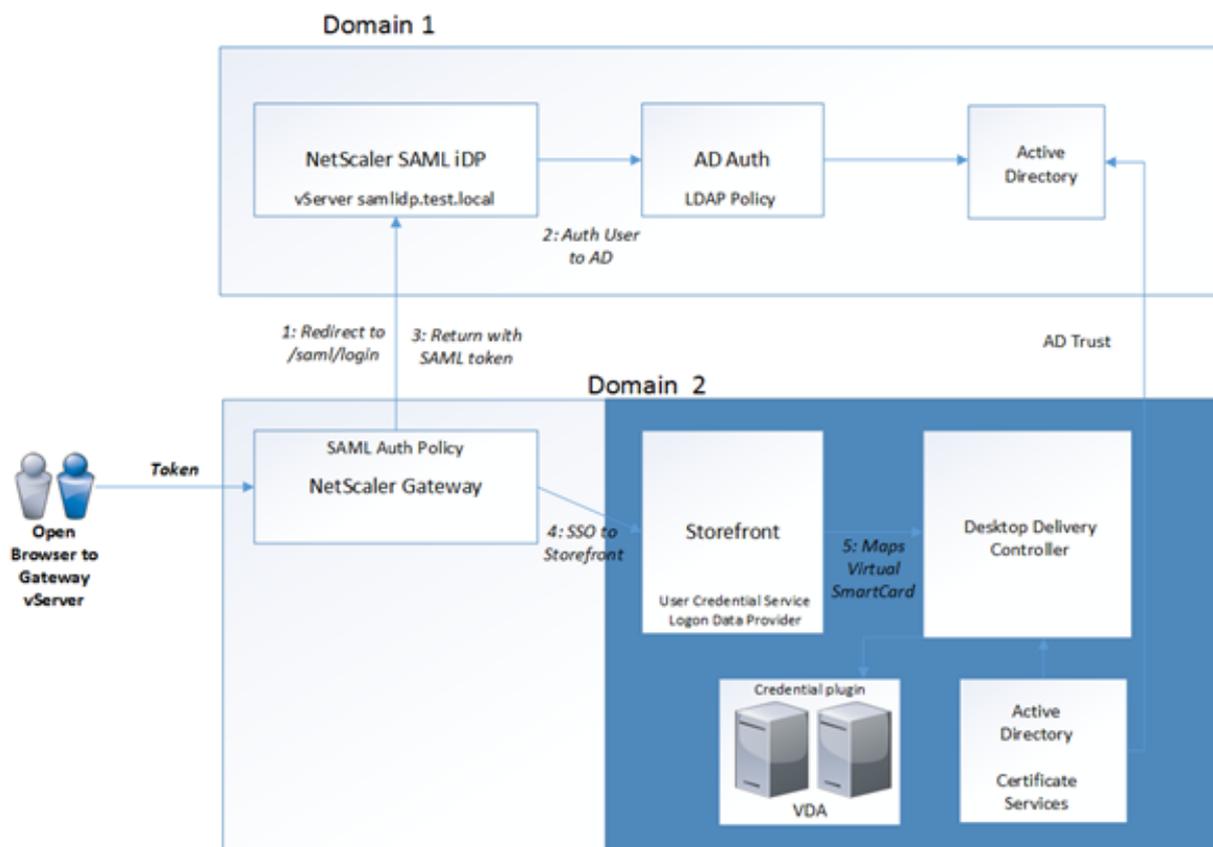
<https://www.citrix.com/blogs/2016/03/03/saml-authentication-technology-preview-for-xenapp-and-xendesktop/>

Now this setup requires that we have an Active Directory certificate services configured in our domain since it issues certificates for use for virtual smart cards to the clients which are connecting, this allows for the SSO feature to happen. It also requires that we have the additional software components called User Credential service installed.

For the simplicity of the setup I have also specified the use of a Netscaler AAA SAML iDP provider. Another option is to use ADFS, but that is dependant on what type of resources we have at our disposal, both options work.

The scenario is described in the screenshot below.

- Now when a user tries to logon the NetScaler Gateway virtual server it will be redirected to SAML iDP based upon the SAML authentication policy. The iDP virtual server has a policy which triggers an AD auth policy and allows for LDAP authentication against the remote Active Directory.
- After auth is successful the SAML assertion is returned to the NetScaler Gateway which then will take the token and apply the session policy and do SSO to Storefront
- Storefront is configured only with NetScaler Gateway pass-through setup and will then see the SAML assertion as a form of Smart Card
- Because of the User Credential Service, Storefront is able to map the SAML identity assertion to convert that into a network virtual smart card logon for active directory.



This solution also requires that we have certificates available on both NetScaler which can be trusted from both ends, typically issues by the same root CA and needs to be trusted by the external users as well.

- Setup an AAA virtual server on the NetScaler appliance in domain two.

- Setup an NetScaler SAML iDP policy on the (samldap.test.local) appliance, according to the screenshot below. This can be done from the GUI under the AAA module.
- Setup an Active Directory authentication policy which points to the local domain controller, this authentication policy should have a higher priority than the SAML policy and have an expression which equals *ns_true*

Note that the nsgw2.test.local is my NetScaler Gateway virtual server which acts as an SAML SP.

Configure Authentication SAML IDP Profile

Name
SAMLIDP

Assertion Consumer Service Url
https://nsgw2.test.local/cgi/samlauth

IDP Certificate Name
wildcard

SP Certificate Name
wildcard

Encrypt Assertion

Encryption Algorithm
AES256

Send Password

Issuer Name
nsgw2.test.local

Service Provider ID

Reject Unsigned Requests

Signature Algorithm*
 RSA-SHA1 RSA-SHA256

Digest Method*
 SHA1 SHA256

SAML Binding*
POST

More

OK Close

- Next we create a policy expression which looks like the screenshot below. This means that if traffic which contains the URL (saml) it should trigger the samldap policy which we just created.

Priority	Policy Name	Expression	Action
100	samlIDP	HTTP.REQUEST.URL.CONTAINS("saml?")	SAMLIDP
101	samlsp2	HTTP.REQUEST.URL.CONTAINS("saml?")	samlsp

- Create an SAML SP policy on the NetScaler Gateway virtual server, this can be done under NetScaler Gateway → Policies → Authentication → SAML. Using the following parameters in the screenshot below.

IDP Certificate Name*
 wildcard +

Redirect URL*
 https://samlidp.test.local/saml/login ?

Single Logout URL

User Field

Signing Certificate Name
 wildcard

Issuer Name
 nsgw2.test.local

Reject Unsigned Assertion*
 OFF

SAML Binding*
 POST

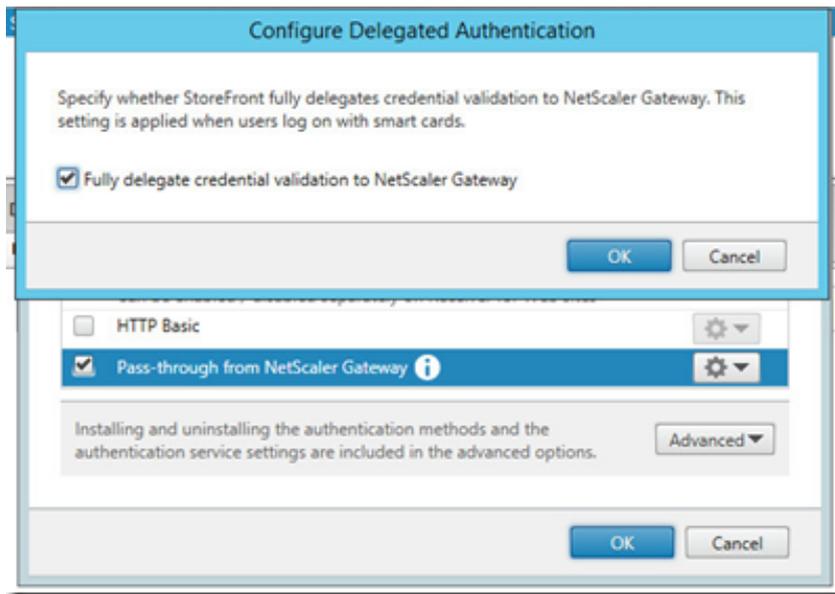
More

OK Close

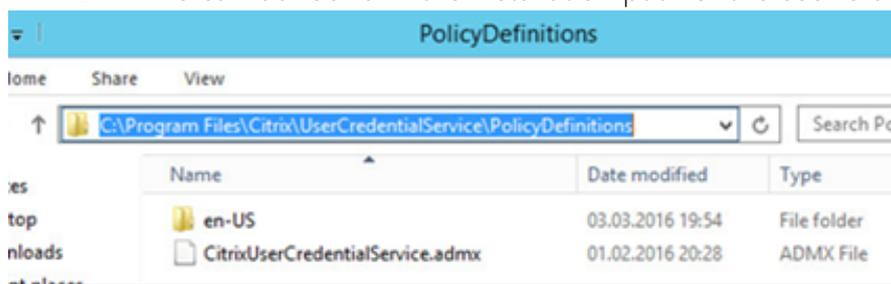
Here we are specifying which URL, the authentication should be triggered on. The URL points to the SAML iDP virtual server on the other NetScaler appliance.

From there we just need to configure a regular session policy as we would normally do to setup redirection to Storefront and configure ICA-proxy settings.

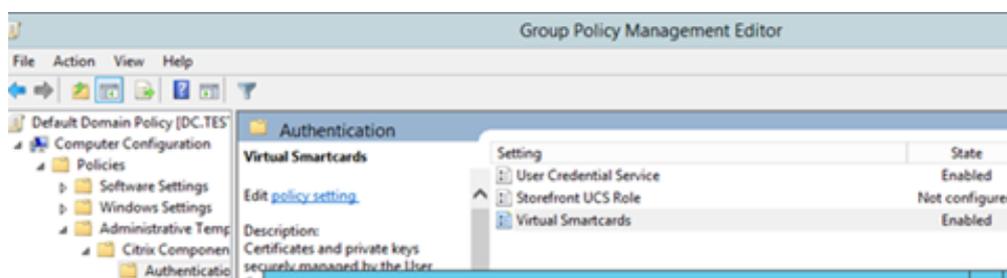
- Now we need to install the User Credential components on another Windows Server (NOTE that you should have configured your Active Directory environment for smart card configuration, the steps can be found here → <http://support.citrix.com/article/CTX206156>)
- Next we need to configure password delegation to the NetScaler from within Storefront



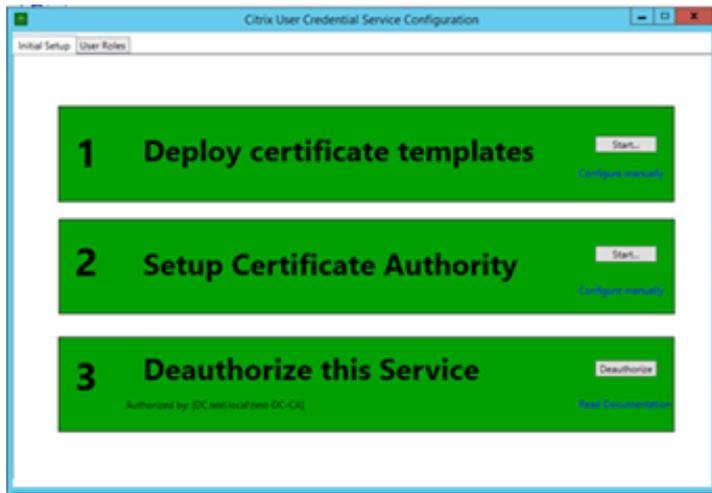
- After we installed the additional components we get an ADMX file which we should place on our central store (domain\SYSVOL\domain\policies\policydefinitions) the ADMX file can be found in the installation path of the user credential service



- We can create a policy using that ADMX template and define settings within the policy. Note that this settings are required for the VDA's which we want to use SSO against and for Storefront server. The two settings which we need to configure are **Virtual Smartcards** and **User Credential Service**. Note under User Credential Service you need to specify the DNS server of the server that runs the UCS service



- After that you need to run a Group Policy update before you continue the process. Next you need to run the Citrix User Credential application on the server which the service was installed on.
- Follow the steps which pop up in the wizard



- After this is complete, we are done with the configuration.

The traffic flow from an end-user will look like this.

So the traffic flow will look like this.

1: Users goes to NetScaler Gateway Virtual Server

2: The NetScaler Gateway Virtual Server SAML Policy redirects logon to SAML iDP logon page

3: SAML iDP has AD authentication policy bound to it, presents an authentication screen to the end-user.

4: After successful login, the SAML iDP redirects SAML assertion back to the SAMLSP which is the NetScaler Gateway Virtual Server.

5: The NetScaler Gateway Virtual Server forwards the SAML Assertion to Storefront

6: Storefront takes the SAML Assertion, communicates with the USC service.

7: UCS service generates an temporary user cert from the CA based upon the SAML assertion.

8: Storefront presents the cert to the VDA agent. The VDA agent looks at the cert which is a trusted cert and gives back authentication approved.

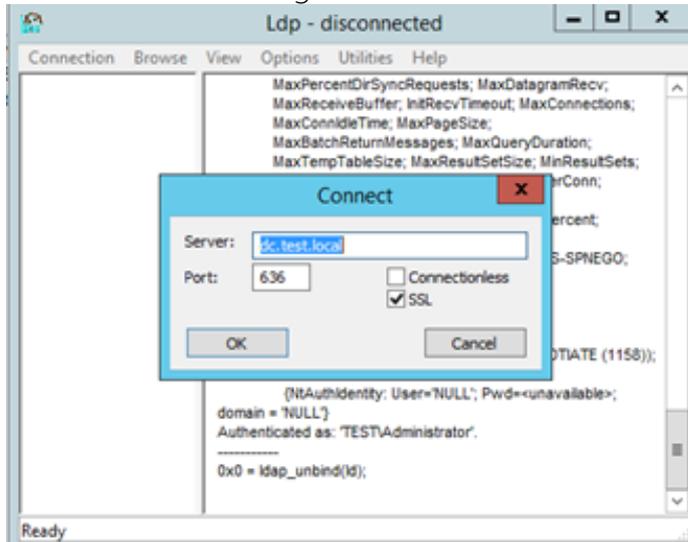
Allow password change from NetScaler Gateway

In order to allow for password, change option to be available from NetScaler Gateway we need to have an SSL/TLS based connection against our domain controllers. This requires that we have a certificate issued to our domain controllers.

- Verify that there is a certificate is installed in the computer's Personal store. If not one should be created and added.
- Start Microsoft Management Console (MMC) Add the Certificates snap-in that manages certificates on the local computer. Expand **Certificates (Local Computer)**, expand **Personal**, and then expand **Certificates**.
- A certificate should exist in the Personal store. In the **Certificate Properties** dialog box, the intended purpose displayed is **Server Authentication**. This certificate is issued to the computer's fully qualified host name.

If you just added a new certificate, then restart the domain controller.

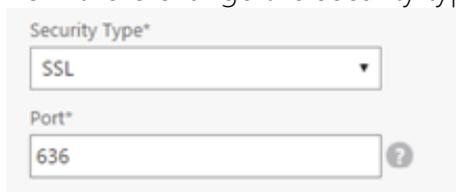
- After this is complete, open up **ldp.exe** and check if you can connect to the controller using SSL



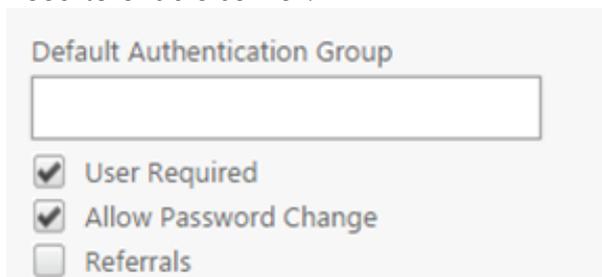
Do this against all domain controllers to verify that it works properly. Now we need to do some changes in our authentication policy on the NetScaler.

Go into the NetScaler gateway virtual server → Authentication → LDAP policy → Edit Server.

From there change the security type to **SSL** and port number to **636**,

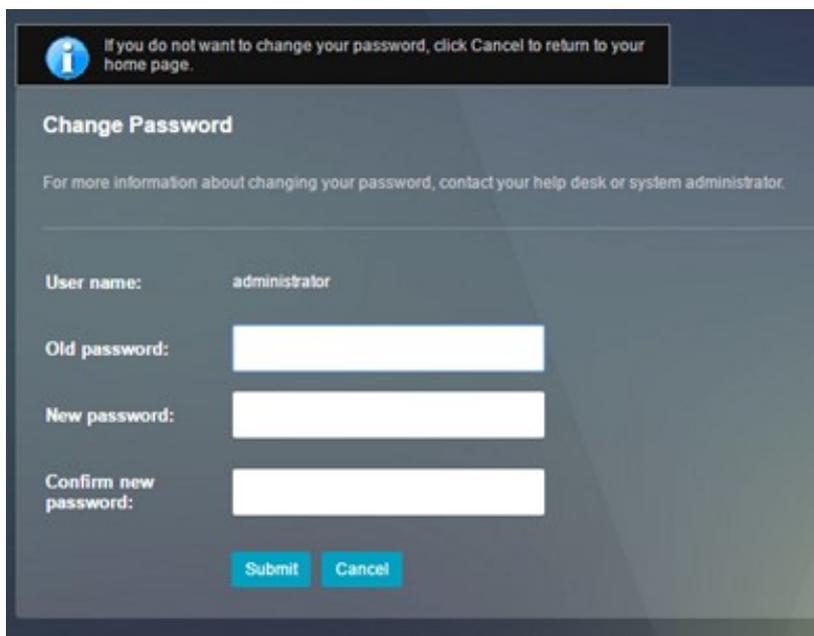
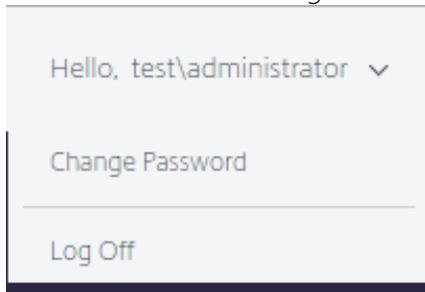


Now further down we will then have an option called **allow password change** which we need to enable as well.



After this is done, we can save our policy and try authenticating again. One thing that should be noted is that this feature **will not work** if we define a new user that should change their password at next logon. It will work is a password expires or if a user wants to change their password manually or if we have an existing user that we them to change their password at next logon.

Clientless access showing



Note that the option to change password directly will not be available from within ICA-proxy mode, but will show option to change password if it has expired when logging in.

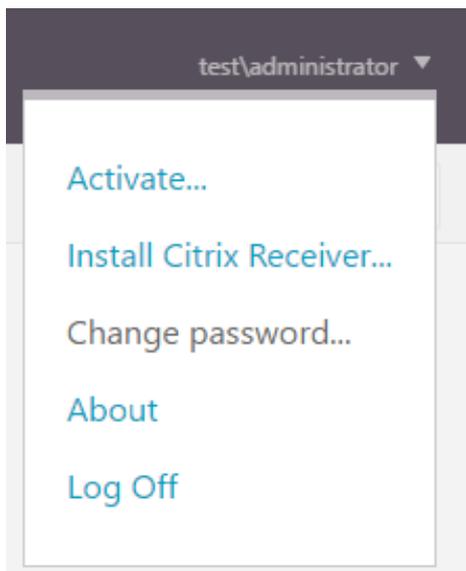
[Allow password change from Storefront](#)

We also have an option to configure password change option directly in Storefront, this is available when users connect using ICA-proxy and allows users to change their password directly from within the portal.

In order to enable this, open Storefront → Manage authentication methods → Pass-through from NetScaler Gateway → Manage password options. From there enable "Allow users to change passwords"



When this is enabled, there will be a new option available for users which login to Storefront using ICA-proxy.



When users change their password here, they will be required to logout and login again.

Multifactor authentication

Multi-factor authentication allows us to enhance the security of end-user authentication which requires that end-user needs to enter another factor besides user-name and password for instance OTP (One-time Password) or PIN-code for instance.

There are multiple two-factor authentication providers such as

- Gemalto Safenet
- RSA
- SMS PassCode
- Azure MFA

For the purpose of this scenario I have chosen Azure MFA to use as an example for setting up MFA for a NetScaler Gateway setup. Most of the different vendors have their own documentation for setting up MFA for NetScaler Gateway.

To simplify the setup, I have chosen to focus mostly on the NetScaler configuration of the MFA configuration, since different vendors have different setup processes and so on.

The traffic flow will look like this.

- An end-user goes to the FQDN of the Virtual Server
- The session applied the Primary Authentication policy, which is bound to the virtual server.
- The session rewrite policy applied which removes the Password 2 menu option at logon
- User enters the username and password for the Active Directory authentication
- Username and password is validated against Active Directory
- After authentication is successful, sessions get the secondary authentication policy bound
- User information is sent to the RADIUS service specified in the secondary authentication policy
- The Azure MFA service validates if the user is configured for two-factor authentication, calls the Azure MFA cloud service
- The Azure MFA cloud service sends SMS with an OTP
- The end-user is sent to the challenge page, and enters the OTP which they received from the Azure MFA cloud service
- After approved authentication, the end-user is forwarded to storefront where they now have access to the resource

To setup this type of scenario we need first to setup Azure MFA component, then configure a user, which has MFA enabled

NOTE: A more detailed setup on the Azure MFA component can be found here → <http://bit.ly/1lkQ0NO>

The should be configured with one-way OTP which means they will receive an SMS on their phone on the phone number that is configured on the user. This OTP they will need to enter on the challenge page to successfully authenticate.

Edit User

administrator

General | **Advanced** | Tags | Administrator | Mobile App Devices | OATH Token | Security Questions | Info

Username:

First name:

Last name:

Email address:

User group:

Country code:

Phone:

Extension:

Enable Global Services

Phone call

Text message

Mobile app

OATH token

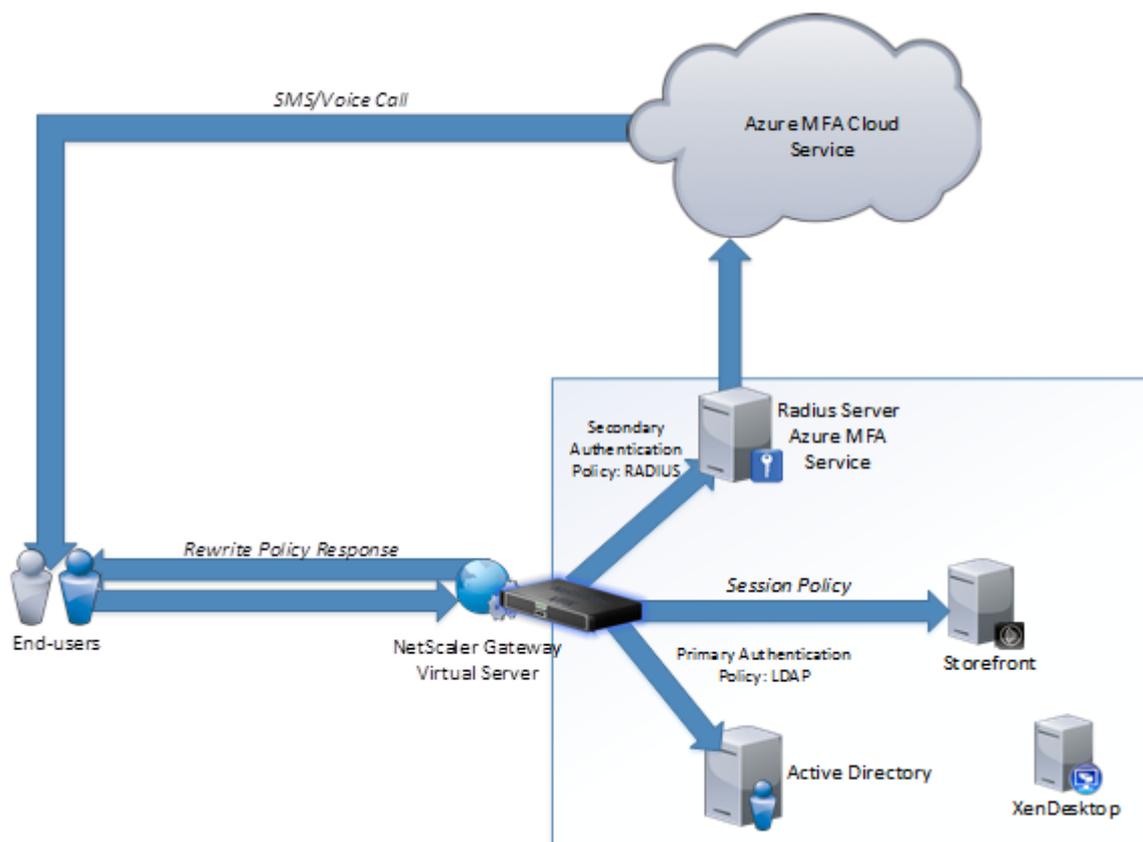
PIN:

Confirm PIN:

Enabled User must change PIN

Send email

Next, we need to configure a rewrite policy, which we bind to the Netscaler Gateway virtual server to remove the Password 2 box, which appears after we add a secondary authentication policy



Go into Appexpert → Rewrite → Actions and click Add. Under type choose INSERT_HTTP_HEADER, under header name write Set-Cookie and under expression type ("pwcount="+ 1")

As shown in the screenshot below

Configure Rewrite Action

Name
INSERT_HTTP_HEADER

Type
INSERT_HTTP_HEADER ▼

Use this action type to insert a header.

Header Name*
Set-Cookie

Expression
Operators ▼ Saved Policy Expressions ▼
("pwcount="+ 1")

Then click OK. Then we need to bind the action to a policy. Go into Rewrite → Policy and click Add. From there give it a name choose the create we just created and under expression enter

HTTP.REQ.HEADER("Set-Cookie").CONTAINS("pwcount").NOT

Then click OK. Now we have to bind this rewrite policy to the NetScaler Gateway Virtual Server. Go into NetScaler Gateway → Virtual Server → Choose an existing virtual server and click Edit. Under policies click the + sign to add another policy.

Under policy list choose rewrite and choose response, then click continue. Then click add binding and bind the existing rewrite policy we just created.

Lastly we need to create an secondary authentication policy which is bound against the RADIUS server. Go back to the virtual server under authentication and click the + sign.

Policies

Choose Policy*
RADIUS ▼

Choose Type*
Secondary ▼

Continue Cancel

From the list choose RADIUS and choose type Secondary, and click Continue. Click add binding and click the + sign to create a new RADIUS authentication policy. Then click + on the server field to add a new RADIUS server.

Add the AzureMFA on-premises service IP, and specify the RADIUS shared secret

Name*
 ?

Server Name Server IP

Server Name*

Port*

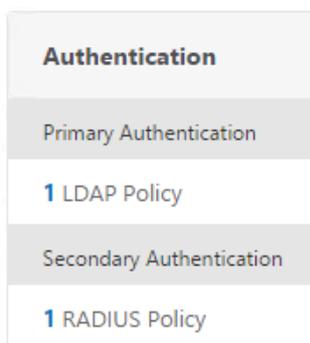
Time-out (seconds)

Secret Key*

Confirm Secret Key*

Then click create. Back to the RADIUS policy enter in the expression field `ns_true` give it a name and choose create, and then to last click Bind.

Now back on the virtual server we should have two authentication policies appearing



When a user tries to authenticate to the NetScaler Gateway virtual server, after they have authenticated using Active Directory credentials. They will receive an SMS containing an OTP and will be redirected to this page, where they need to enter their OTP in order to authenticate successfully.

Additional Information Required

Please type your response below.

NetScaler with
Unified Gateway

Enter the verification code you receive via text message.

Submit

NOTE: There is a separate KB article on how to remove the Password 2 option for Citrix Receiver connection, which can be found here → <http://bit.ly/1Q3ntXq>

Authorization

When we configure an authorization policy, we can set it to allow or deny access to network resources for example, after we have configured a full VPN solution and we do not want access to resources besides those we have configured. Authorization policies can be bound to an AAA user or an AAA group.

So as an example, we configure a full VPN without the use of split tunneling, when that is configured, users will be treated as an internal connected user, and then have access to all internal resources if not any firewall rules have been configured. This is where we can use authorization policies to ALLOW/DENY access to particular resources or networks.

Now in order to create authorization rules go into NetScaler Gateway → Policies → Authorization → Add, from there you specify a name for the rule and then define an action ALLOW or DENY, which will be triggered if the expression is evaluated to be true. Therefore, as an example, if we want to DENY access to a particular resource based upon IP-address we can use the following expression.

```
REQ.IP.DESTIP == 10.217.215.107
```

In addition, specify DENY action.

Create Authorization Policy

Name*
DENY_IP

Action*
DENY

Operators Saved Policy Expressions Frequently Used Expressions

REQ.IP.DESTIP == 10.217.215.107

[Switch to Default Syntax](#)

After we have created the rule, we have to bind it, either to a particular AAA user or an AAA group. Go into NetScaler Gateway → User Administration → AAA Groups → Click edit on an existing group or create a new one. Go into Authorization policies and click on the “No Authorization Policies” button then choose “Click to select” and select the pre created policy and click Bind and OK.

AAA Group

Basic Settings

Group Name
vpx

Authorization Policies

1 Authorization Policy

Done

NOTE: When creating AAA group you need to specify the name of the Active directory group you want to filter on, and be aware that it is case sensitive.

So now when a user from the particular group logs on the VPN client using the NetScaler gateway plugin they will be unable to connect to that particular address as shown from the connection log of the Gateway plugin.

Time Stamp	Data
Tue Apr 05 14:42:05	The connection to the server 10.217.215.107:80 failed
Tue Apr 05 14:42:05	The TCP port deleted from 192.168.101.5:51104 to 10.217.215.107:80
Tue Apr 05 14:42:05	The TCP connection to 10.217.215.107:80 is denied.
Tue Apr 05 14:42:05	The connection to the server 10.217.215.107:80 failed
Tue Apr 05 14:42:05	The TCP port deleted from 192.168.101.5:51105 to 10.217.215.107:80
Tue Apr 05 14:42:05	The connection 192.168.101.5:11837 to 199.16.156.6:443 is establis...
Tue Apr 05 14:42:05	The connection 192.168.101.5:11837 to 199.16.156.6:443 is establis...
Tue Apr 05 14:42:06	Processing a request for a tunnel from 192.168.101.5:51108 to 10.21...
Tue Apr 05 14:42:06	Making a connection to 10.217.215.107:80 by chrome.exe ...
Tue Apr 05 14:42:06	The TCP connection to 10.217.215.107:80 is denied.

Now in most cases we might have multiple authorization policies per group and even some policies bound to the AAA user and therefore it is important to understand the order of policy processing and which. The tables below shows some examples and how they are processed

ACTION	EXPRESSION	BOUND	PRIORITY
ALLOW	REQ.IP.DESTIP == 10.217.215.107	AAA Group REMOTE	80
DENY	REQ.IP.DESTIP == 10.217.215.107	AAA Group REMOTE	100

For this, all users, which are part of the group REMOTE, will get access to the IP 10.217.215.107. This is because those policies, which the highest number will be, processed first and then going upwards.

ACTION	EXPRESSION	BOUND	PRIORITY
DENY	REQ.IP.DESTIP == 10.217.215.107	AAA Group REMOTE	80
ALLOW	REQ.IP.DESTIP == 10.217.215.107	AAA User John	70
ALLOW	REQ.IP.DESTIP == 10.217.215.107	AAA Group REMOTE	100
ALLOW	REQ.IP.DESTIP == 10.217.215.107	AAA User Jeff	81

In this scenario, we added an AAA user as well. With this setup policies will be processed first with the ALLOW RULE for group Remote, then DENY rule for group Remote. Now John will get access to that resources since his allow policy has a lower priority then the DENY rule for AAA group remote. Jeff also has an ALLOW policy but it has a higher number then the DENY rule and will therefore not get access. So AAA User and AAA groups policies are handled the same way, therefore a common practice is to use AAA group policy priorities from 100 – 200 for instance and user policies to be between 10 – 100 to avoid conflicting policies between groups and users.

Now when setting up authorization policies we have a bunch of different types of expressions we can use in order to filter based upon the endpoint or different network properties.

We can for instance use OPSWAT to do filtering based upon the security or patch level of the endpoint, but we can also use general networking filters such as

- IP: SOURCEIP, DESTIP
- HTTP: METHOD, URL,, VERSION
- TCP:SOURCEPORT, DESTPORT
- SSL:CLIENT.CERT, CLIENTCERTISSUER

In addition, we can combine these settings to do more granular filtering, so for instance if we want to deny access to a particular IP on a particular port an expression would look like this.

ACTION DENY

REQ.IP.DESTIP == 10.10.10.10 && REQ.TCP.DESTPORT == 80

Where the && = AND, we can also use || = Or attribute in the expression, but in this case it needs to be a connection to 10.10.10.10 port 80 in order for the expression to be evaluated to be true and then deny the connection.

Troubleshooting

This section contains a list of troubleshooting issues and how to resolve them, and is categories within each feature.

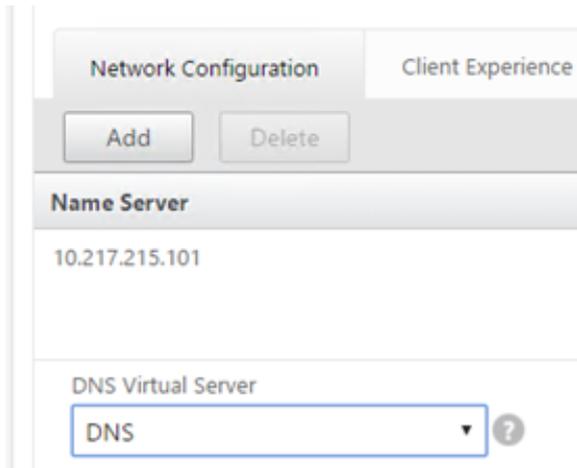
Endpoint Access

Name resolution not working

After a user has connected using Endpoint Access and is unable to resolve names, this can be of name servers not properly defined on the NetScaler.

By default, all we need it to configure a global name server on the NetScaler. Which is done in the GUI under Traffic Management → DNS → Name Servers. Now if we are using endpoint access and are given an IP address which cannot reach this DNS server we cannot do DNS queries, in that we need to specify a DNS virtual server under the global settings of the NetScaler Gateway. This can be setup as a simple load balanced DNS virtual server which is then placed on a subnet where the endpoints have access to it.

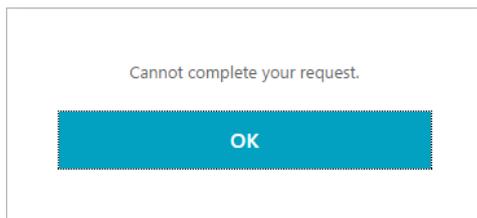
To configure this, go into NetScaler Gateway → Global Settings → Network configuration → DNS Virtual Server. Note that this will ONLY list available virtual servers which are configured with the DNS protocol.



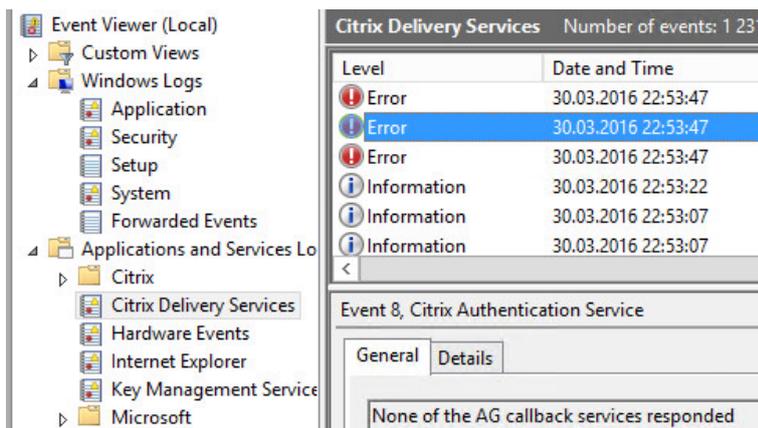
ICA-proxy

Cannot complete your request

After logging into the NetScaler Gateway and the enduser is redirected to the StoreFront page you get the error message "Cannot Complete your request"



You can also notice that you get an error in event viewer of the storefront server under Application and Services Logs -> Citrix Delivery Services. Where you get an error message of "None of the AG Call back service responded"



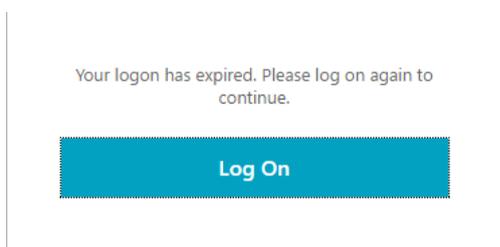
This is often the case if Storefront cannot talk back with the callback URL which is listed under Manage NetScaler Gateways → Edit NetScaler Gateway → Authentication Settings → Callback URL. Make sure that this URL is accessible from the Storefront server. If this is not possible because of network segmentation. You can deploy a dummy NetScaler Gateway VIP in the internal network.

If you notice that you have an error in Event viewer stating that *"Citrix AGBasic Login request has failed"*. That might be that there are different domains specified on the NetScaler session policy and under Storefront. If you have specified a domain name in Storefront under **Manage Authentication → Pass-through from NetScaler Gateway → Configure trusted domains**, this needs to be the same domain name in the session policy as well.

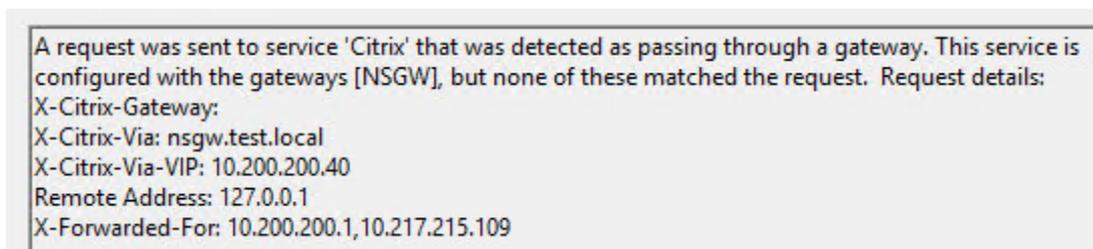
If you note that you have an error in Event viewer stating that *Failed to run discovery* this is most likely the case if you have not configured the use of a proper SSL certificate under the IIS administration console of the Storefront server.

Your logon has expired

If you are prompted for another authentication after logging into the NetScaler Gateway portal, when redirected to the Storefront portal, and then this error message appears.



You can also notice an error in event viewer of the storefront server under **Application and Services Logs -> Citrix Delivery Services**. That states, "A request was sent to service that was detected as passing through a gateway, but none of these matched the request."



This is typically the case if the NetScaler Gateway URL is configured wrongly. Since this URL needs to be the same as what the end-users are using, in case Storefront will not trust the incoming request and therefore ignore authentication attempts.

Display name:	<input type="text" value="NSGW"/>
NetScaler Gateway URL:	<input type="text" value="https://nsgw.test.local"/>
Usage or role: ⓘ	<input type="text" value="Authentication and HDX routing"/>

Unknown Client error 1110

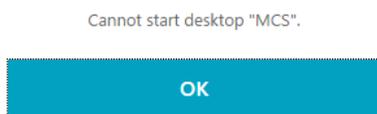
This is a generic error which might occur in many different scenarios, but some key things to check to find the root cause of the issue.

- STA available on the NetScaler and marked as up? (This can be checked under NetScaler Gateway → Virtual Server → Published Applications → STA Server.

VPN Virtual Server STA Server Binding		
<input type="button" value="Add Binding"/> <input type="button" value="Unbind"/>		
Secure Ticket Authority Server	Secure Ticket Authority Server Address Type	State
http://10.217.215.107	IPV4	● Up
<input type="button" value="Close"/>		

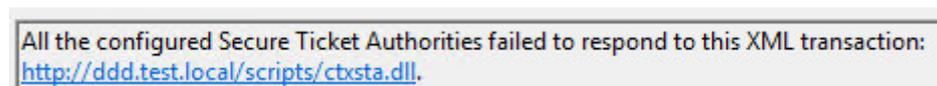
Cannot Start Desktop "COMPUTERNAME"

If you try launching an application or desktop and you get the error message cannot start Desktop/Application name after authenticating and getting the resources up



This might just be that the resource that we are trying to launch is currently unavailable or that something for instance is wrong with the VDA agents on that resource we are trying to launch.

We can also go into event viewer of Storefront to take a closer look at what kind of error is actually happening. Event viewer → **Application and Services Logs** → **Citrix Delivery Services**. If we get an error message here stating "All the configured Secure Ticket authorities failed to respond"



This might be that we have an STA server that is down, in which Storefront tries to communicate with or that we have configured the wrong STA server under NetScaler Gateway appliances in Storefront. This can be checked under → Manage NetScaler Gateways → Edit NetScaler Gateway → Secure Ticket Authority.

Error: Login exceeds maximum allowed users

When logging in you get an error message stating that login exceeds maximum allowed users. This is typically the case if we did not place the virtual server in ICA-only mode. By default, the global AAA settings of NetScaler Gateway is set to allow maximum 5 users logging in using VPN at the same time. If we go and **change the settings of the Virtual server to ICA-only mode**, this error will go away.

Http/1.1 Internal Server Error 43531

After authenticating to the NetScaler Gateway portal you get a blank page with an error message stating Http/1.1 Internal Server Error 43531. This is typically the case if the Gateway cannot communicate with the Storefront web site. Which might just be a wrong URL in the session policy for instance.

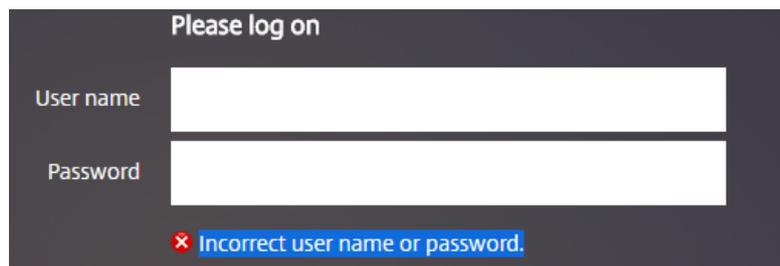
Or this can also be that a client is not being applied a session policy, if we for instance have session policies in place based upon different criteria's. If someone outside those criteria would get those error messages. The easiest way to get them access is to bind a session policy with the highest priority number with an expression of `ns_true`.

403 - Forbidden: Access is denied

After authenticating to the NetScaler Gateway portal, you get a default IIS error message stating "Access is denied". This is typically the case if the session policy does not point directly to the receiver for web site on Storefront. After changing, the session policy to point to the direct URL this error message will go away.

Authentication

In case of authentication failure attempt, a user will be given a generic error message of:



The image shows a dark-themed login form with the title "Please log on". It contains two input fields: "User name" and "Password". Below the fields, there is a red error message: "Incorrect user name or password." The error message is highlighted with a blue selection box.

There are many ways to troubleshoot authentication failures, the simplest one is using the authentication dashboard in the NetScaler UI.



Which basically list the syslog events directly into the UI. Another way is using CLI. Log into the NetScaler appliance using an SSH client, type **Shell** and then type **cat /tmp/aaad.debug**

This will in real-time list out all AAA attempts happening against the NetScaler. Now by default the NetScaler does not list out detailed information whenever a user has an expired password or if their account is disabled. However, there is a feature which we can

enabled which can give more detailed information back to the end user. This feature is called **Enhanced Authentication Feedback**

Default Authentication Type*

LOCAL ▼

Enable Static Caching

Enable Enhanced Authentication Feedback

Enable Session Stickiness

Which enabled under **NetScaler Gateway** → **Global Settings** → **Change Authentication AAA settings**.

NOTE: This setting is disabled by default, because it might reveal too much information to malicious hackers which try to do a brute force attack, to get information on which users are enabled and not.

It is also important that the `aaad.debug` command lists out different error codes when there is a failed authentication attempt.

For instance, if a user with a disabled account tries to authenticate.

Send reject with code Rejecting with error code 4011

```
Fri Apr 1 14:28:17 2016
/home/build/rs_110_64_24_RTM/usr.src/netscaler/aaad/ldap_drv.c[395]: receive_ldap_user_search_event User DN= <<CN=test5,CN=Users,DC=test,DC=local>>
Fri Apr 1 14:28:17 2016
/home/build/rs_110_64_24_RTM/usr.src/netscaler/aaad/naaad.c[2583]: send reject with code Rejecting with error code 4011
```

Citrix has made a list which describes all these error codes and the meaning of them.

- 4001 Invalid credentials. Catch-all error from previous versions.
- 4002 Login not permitted. Catch-all error from previous versions.
- 4003 Server timeout
- 4004 System error
- 4005 Socket error talking to authentication server
- 4006 Bad (format) user passed to nsaaad
- 4007 Bad (format) password passed to nsaaad
- 4008 Password mismatch (when entering new password)
- 4009 User not found
- 4010 Restricted login hours
- 4011 Account disabled

- 4012 Password expired
- 4013 No dial-in permission (RADIUS specific)
- 4014 Error changing password
- 4015 Account locked

Now if a user tries to authenticate but is not bound to an authentication policy, for instance if we have multiple authentication policy for different groups, network segments and someone which fall outside of those policies try to authenticate they are presented with this error message.

**No active policy is found in Primary authentication cascade
Please contact your administrator.**

The simplest way to fix this is to either define **ns_true** authentication policy which handles all other authentication attempts.

Now if an end-user tries to authenticate to start a Citrix Receiver session and is presented with this error message

Error: Not a privileged User.

This is typically the case if there is a session policy bound to the user which has a default authorization policy of **DENY**, this might be intended but if not, we should change it to **ALLOW**.

Other design examples

Multitenant ICA-Proxy

This scenario deals with the issue if we have multiple tenants which needs to connect using a single IP address to a set of different XenApp/XenDesktop farms, which also live, inside their own Active Directory domain.

NOTE: This scenario cannot be done using a regular NetScaler Gateway VPX but requires a NetScaler either standard or higher.

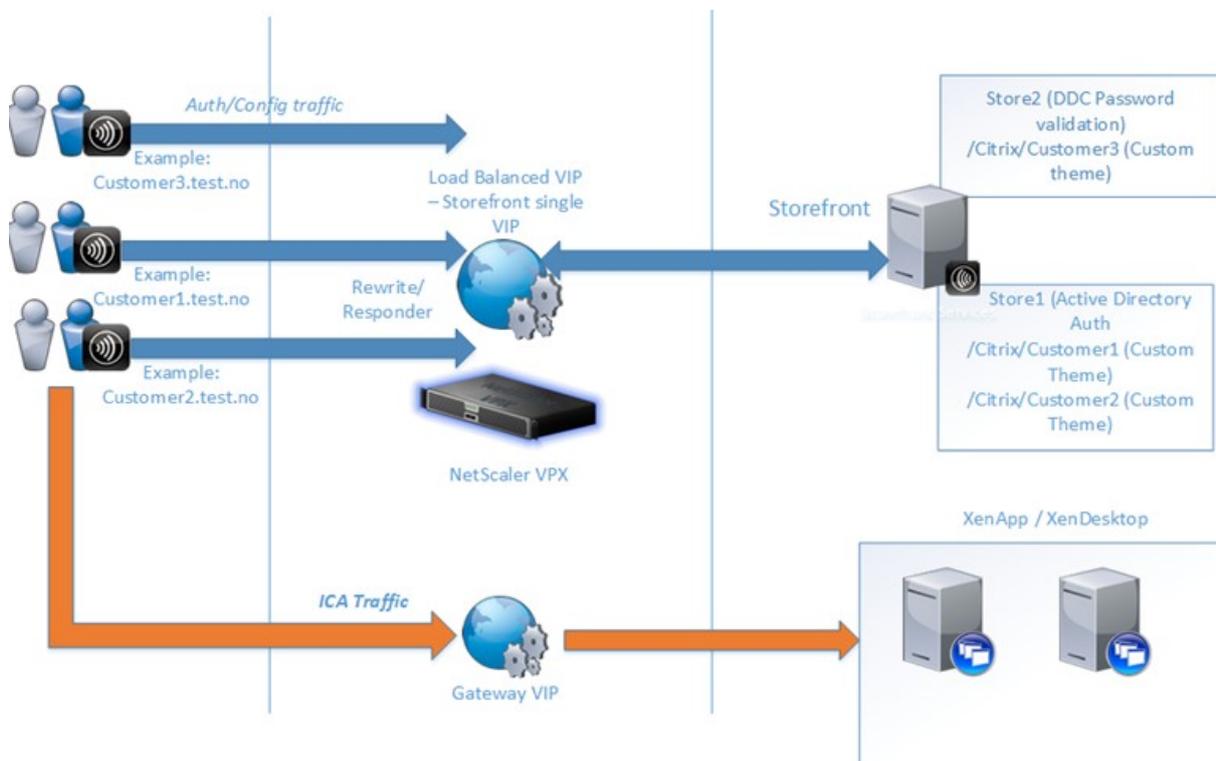
We achieve this by

- We publish Storefront as a LB virtual server behind the NetScaler (Meaning that Storefront is accessible from the external network)
- We configure a Gateway virtual server, which will handle the ICA traffic.
- We use Responder, Rewrite policies to handle the redirect to the correct URLs.
- We configure Optimal Gateway Routing with direct access on Storefront (Which means that all ICA traffic regardless of beacons will be redirected using a Gateway.

This feature is not new, but with Storefront 3.5 this is available in the GUI. We also define that Gateways are being used for HDX routing only, all authentication will happen on Storefront.

- We have one or multiple Storefront stores depending on the requirements for backend setup for instance if we have multiple isolated active directory, and we have defined password verification against DDCs instead of Active directory. This might vary from deployment to deployment but important to remember what are Store specific settings and what are Receiver for web specific settings.
- We can have multiple Gateway virtual servers to handle communication, but customers still only need one virtual server for the external Storefront portal.

In the screenshot below is the overview architecture:



User flow:

When a user starts receiver for the first time and tries to configure his Receiver it will be communicating directly with the Storefront endpoint and configures properly.

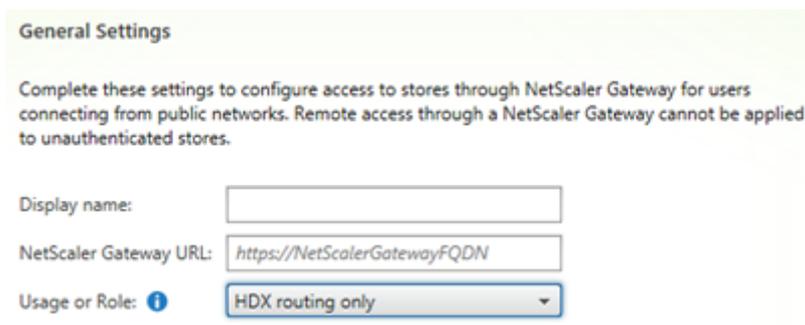
Depending on what kind of Store the user is, accessing this might be done using DDC validation or using Active Directory. Same goes if using Receiver for web, the user connects and is typing his customer name is redirected to the customer website on Storefront. When the user tries to start an application or desktop session, the session will generate an ICA file containing the Optimal Gateway setting the session will be routed using the Gateway.

Setting up:

- Setup a load balanced virtual server for Storefront
- Setup a generic Store(s) in Storefront (Depending on the requirements we can have one or multiple Stores, which again link to its own Storefront web site. Authentication is handled inside each individual store. Example is having two stores, which links to two tenants.
- Setup the base URL of Storefront using SSL (The URL here does not matter since it is not shown for the end-user)
- Create Receiver for web sites for each store or multiple sites for one store depending on the requirements we have for authentication
- Create a NetScaler Gateway Virtual Server which is set to ICA-only mode, disable authentication, add an SSL certificate and STA server.

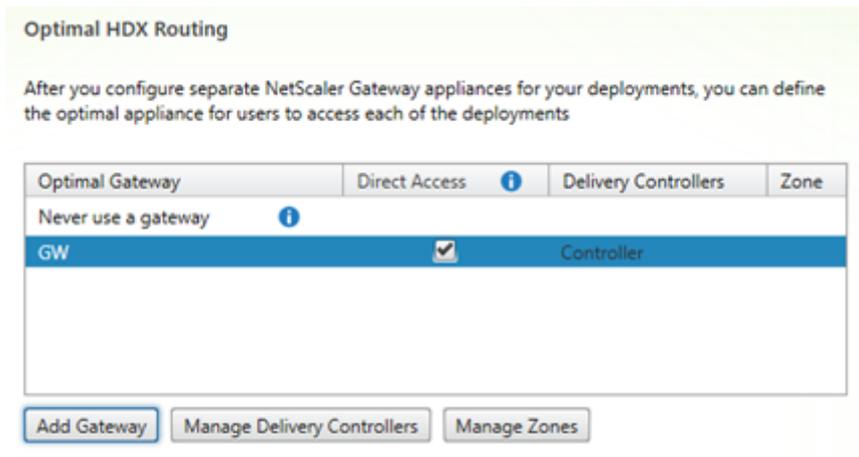
Note: We can alter what we want for each website, portal customization can be done under the `c:\inetpub\wwwroot\citrix\<(nameofwebsite)` or using the Storefront GUI.

- First, we need to add a NetScaler Gateway in Storefront, when doing so we need to define that it should only be used for HDX Routing and not for authentication. The FQDN here should be the NetScaler Gateway virtual server, which will handle all the ICA-sessions.

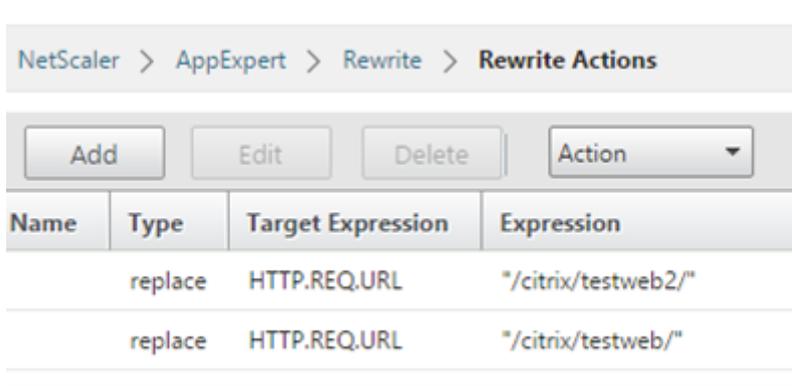


The screenshot shows the 'General Settings' configuration page for a NetScaler Gateway. The page has a light green header with the title 'General Settings'. Below the header, there is a descriptive paragraph: 'Complete these settings to configure access to stores through NetScaler Gateway for users connecting from public networks. Remote access through a NetScaler Gateway cannot be applied to unauthenticated stores.' There are three input fields: 'Display name:' with an empty text box; 'NetScaler Gateway URL:' with a text box containing 'https://NetScalerGatewayFQDN'; and 'Usage or Role:' with a dropdown menu set to 'HDX routing only' and a blue information icon to its left.

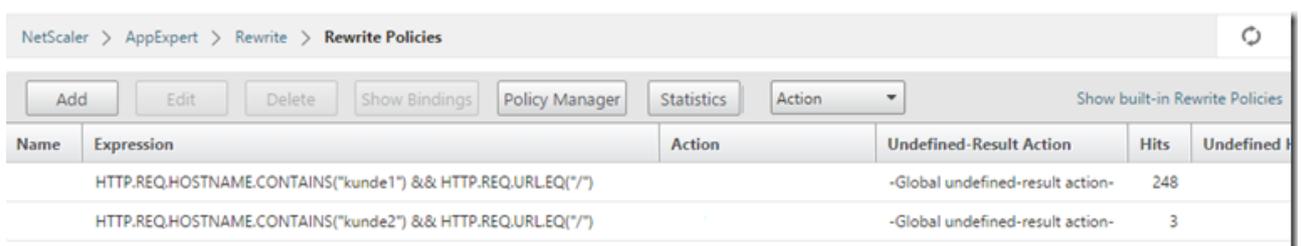
- Now we need to define use of that Gateway within the Store, this can be done by going into the Store settings → Optimal HDX Routing (Storefront does not need to be able to communicate with the Gateway, since Authentication is done completely at Storefront. After you have added the gateway, click for Direct Access and define which controllers should be used against the optimal gateway)



- Lastly we need to define some rewrite rules on the NetScaler so that each customer is redirected to their own Receiver for web site. Go into the NetScaler GUI → AppExpert → Rewrite → Rewrite Policies → Add. From there give a name for each policy depending on the customer name.
- Rewrite rules: these are pretty simple just replaces a URL prefix at the end



Then we have an expression that looks at the host name and specifies that the URL must be at the root to it continue



These policies need to be added to the Storefront LB virtual server.

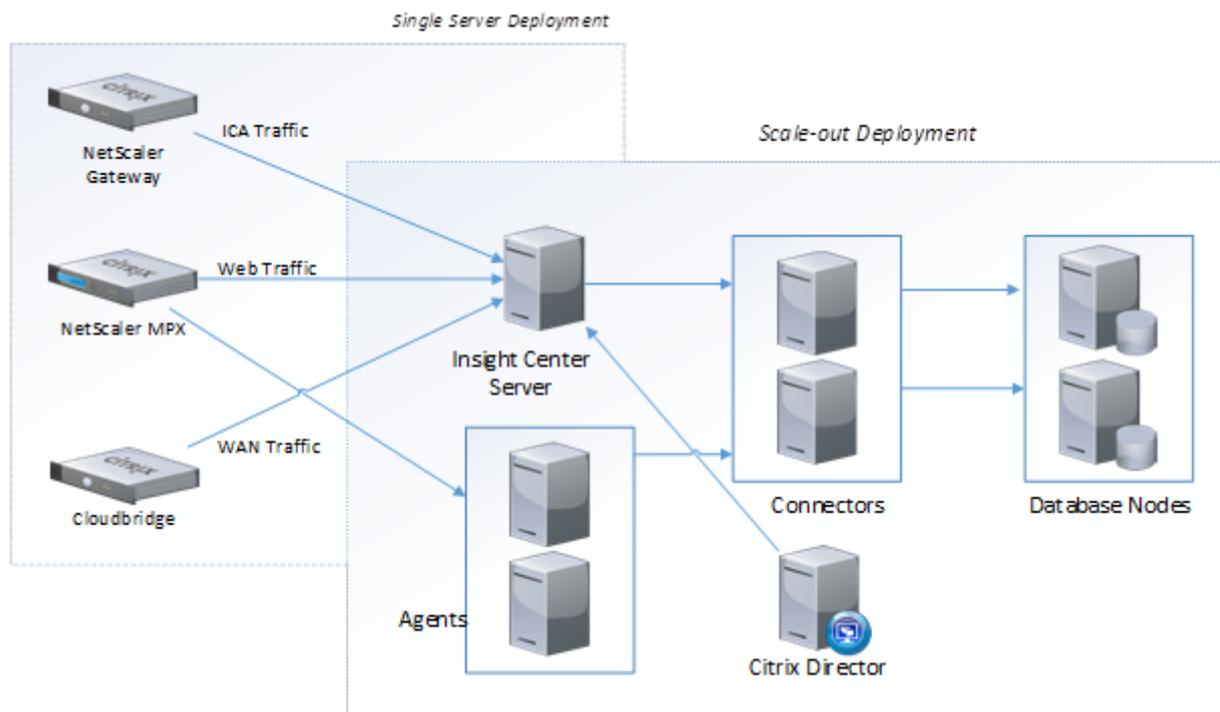
Therefore, when a customer enters their FQDN for the portal site they will be processed by the rewrite policy for the load balanced storefront server to their own Storefront receiver for web. The Storefront load balanced virtual server will handle so all HTTP traffic, and when a user clicks on an application the NetScaler Gateway virtual server, which is specified in StoreFront for HDX routing, will handle it.

Monitoring

This Section is more about how we can monitor our NetScaler infrastructure using different tools that are available. This is not meant as an in-depth part on how to setup and configure the different options, but more on showing what they are capable of.

Insight Center

NetScaler Insight is a virtual appliance, which allows collection of information and data using AppFlow and using that data can do reporting and monitoring based upon different criteria such as Citrix HDX Insight and Web Insight. Insight Center can be deployed either in a single server deployment where we just use one virtual appliance which is deployed to any of the supported Hypervisors, or if we require more logging and higher amount of data retention we can setup a scale out deployment which uses multiple virtual appliances to scale out the required infrastructure.



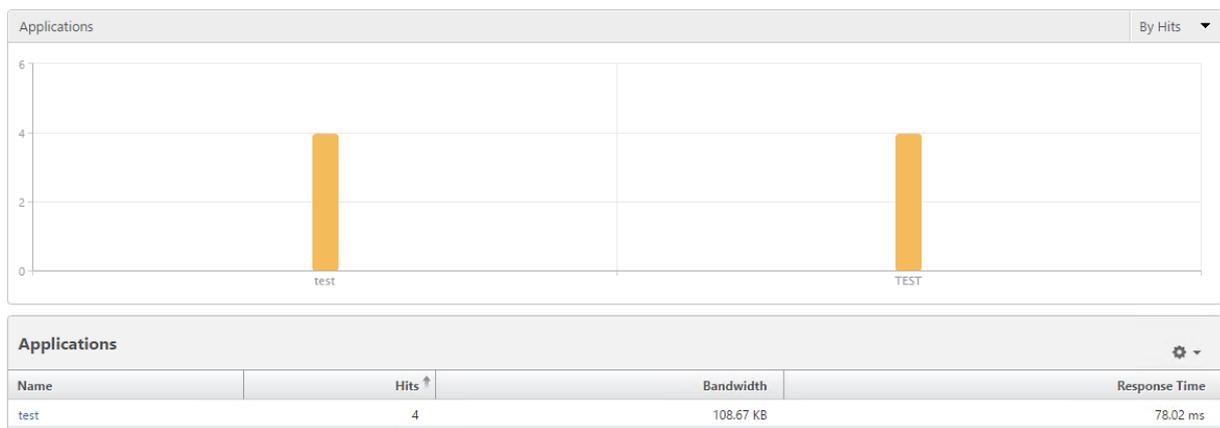
From there we have different roles. Agents, which are used as an AppFlow Collector, which gathered data from the different NetScaler appliances, important to note that as of now Agents, only, support gathering of HTTP AppFlow data. Insight Center is the master server for the deployment and is the one that will gather ICA traffic information using AppFlow. Insight Center Server is the only appliance needed in a single server deployment. Both the agents and the Insight Center Server will communicate with the Connectors, which will in turn distribute the data across different database nodes in the back. We can also integrate Insight with Citrix Director, which will give us detailed ICA network information from Insight directly in Director.

NOTE: All the different roles are included in the same virtual appliance. We choose which role the NetScaler Insight server should have when we setup the virtual appliance for the first time.

The monitoring feature in Citrix Insight is broken down into five part.

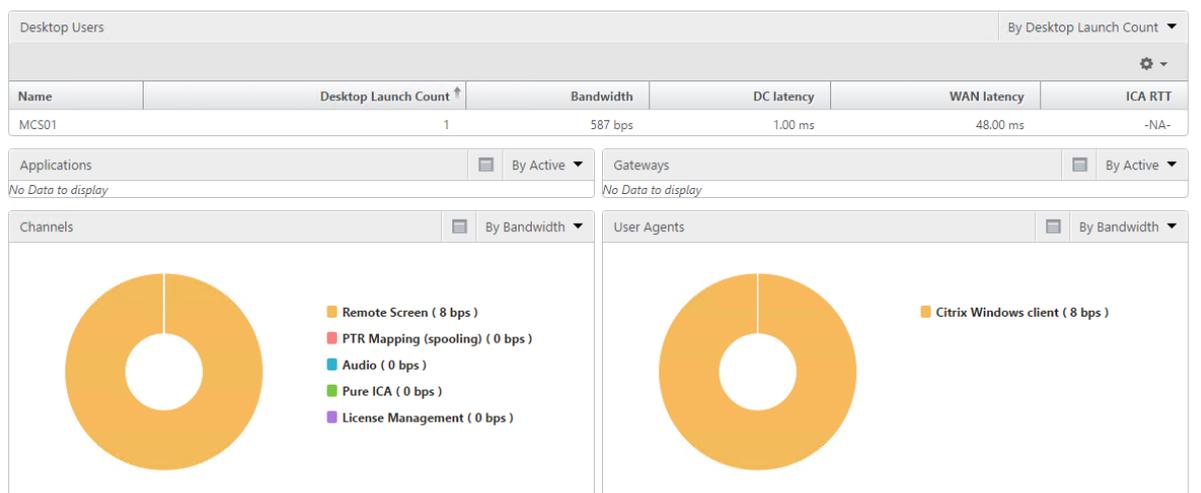
- Web Insight
- HDX Insight
- WAN Insight
- Gateway Insight (New as of build 11.65)
- Security Insight (New as of build 11.65)

Web Insight collects information about load balanced virtual server or content switching servers and can display statistics and information regarding connections



It can display information such as clients, servers, applications (virtual servers), bandwidth usage, and user agents and so on.

HDX Insight gathers ICA session information and can display information such as endpoint, client RTT, WAN latency and so on



In order to use this feature it requires a certain license level on the NetScaler. HDX Insight data is only stored for 31 days even if you have platinum license. If you have NetScaler standard you will only get real time statistics and no option to set retention time. If you have enterprise, you get one hour of data. If you have platinum you have one month of data. If you want to integrate Insight with Director as well, you need XenDesktop/XenApp platinum as well. However, take note that

HDX insight is only for ICA-proxy session it will not display any information regarding to clientless access or FVPN connections.

WAN Insight displays information gathered from Cloud bridge devices. It can display detailed information such as, compression ratio, active accelerated connections, average RTT and so on.

Gateway Insight displays information related to clientless VPN and full VPN connection. It will also display information related to endpoint analysis.

Lastly we have Security Insight which displays information in regards to security settings on the NetScaler.

Command Center

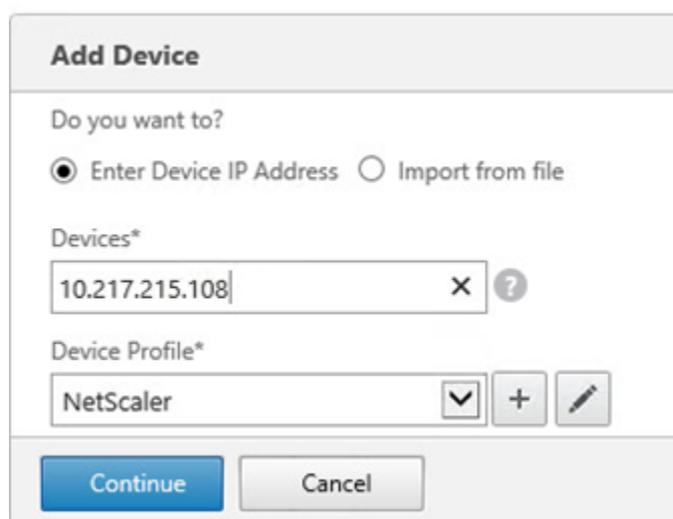
Command Center is a web based management software from Citrix, which can be used to manage NetScaler Gateway, NetScaler, NetScaler Appfirewall & Cloud bridge and so on

We can install the Command Center server on either the Windows or Linux platform. You can download the installation package for both Windows and Linux from the Citrix portal Web site: <http://mycitrix.com>

It can be installed using either a MSSQL database or using an internal PostgreSQL database. PostgreSQL is chosen if we want to install an evaluation of the software.

The default username and password is **root and public**, you will be asked to change to another password after logon.

After logging into and changing password, you will be given the option to add an appliance to manage.



Add Device

Do you want to?

Enter Device IP Address Import from file

Devices*

10.217.215.108 X ?

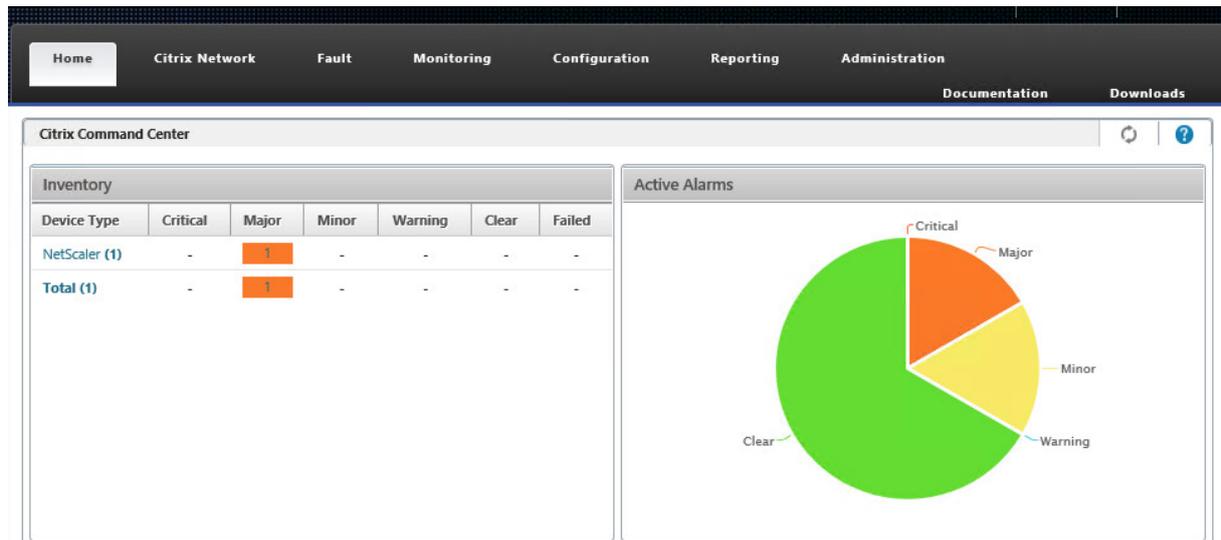
Device Profile*

NetScaler [v] + [edit]

Continue Cancel

Now from within Command Center we can use that to do service and virtual server monitoring. We can also use it to execute multiple tasks in parallel against multiple NetScaler instances. We can also define our own SNMP triggers and then from there use those triggers to run a custom tasks to do email notifications or to run custom commands and so on.

So from within the dashboard we get a good overview of the current status of each system



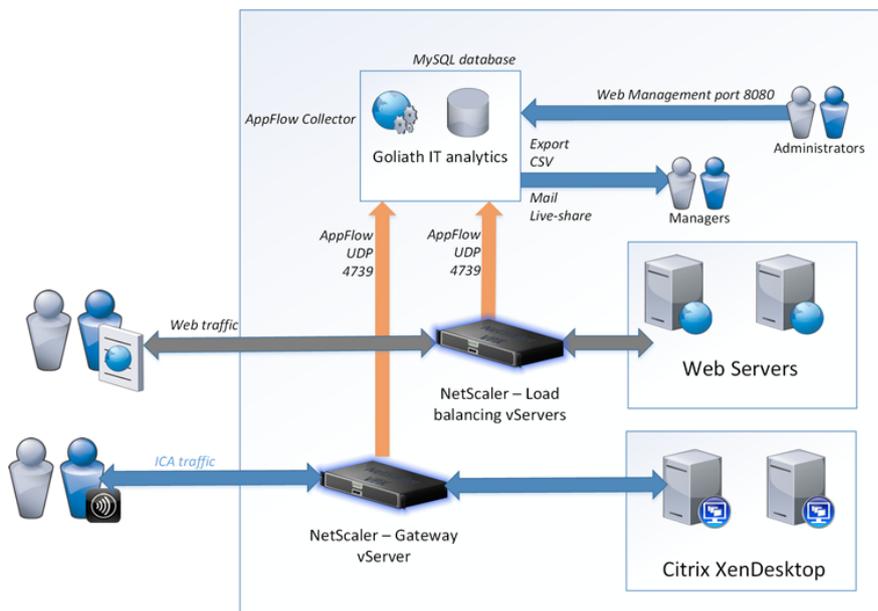
Based upon SNMP and Syslog events.

Goliath IT analytics for NetScaler

Goliath IT analytics for NetScaler falls within the same category as NetScaler Insight, since it also uses the AppFlow engine to gather data. You can download a trial here →

<http://goliathtechnologies.com/software/goliath-for-netscaler/#->

However, it has a lot simpler architecture and doesn't share the same limitations that Insight has in terms of data retention and in terms of total connections. All the data that is gathered is stored within a MySQL database within a virtual appliance. This also gives us a lot of flexibility since we can then do what we want with the data as long as we are a bit familiar with MySQL.

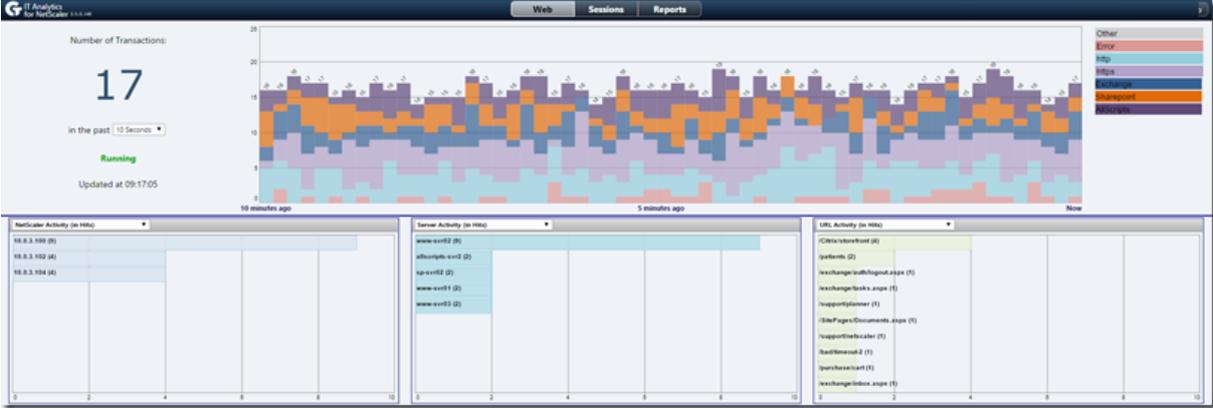


Goliath can run as a virtual appliance on any hypervisor (VMware, XenServer or Hyper-V) and one of the main features, it is that you can pretty much store data for as long as you want to. If we think about it this gives us the benefit to measure the results of optimization changes. For instance, we can compare RTT and average latency of a NetScaler gateway session for the last 30 days and the compare from month to month after we have adjusted the TCP settings on the NetScaler's.

From the main dashboard, I can also easily see which URL's are often accessed. This view can also give me a good indication if something is trying to brute force login to a particular URL.



I can also get a quick overview on what kind of traffic is coming, in for instance see what the server activity is like and which NetScaler's have active connections. I can also see the total concurrent transactions happening in real-time



And also like with NetScaler insights I also have HDX insight capabilities which allows us to see all the Citrix ICA session directly within the tool

User IP	Version	Started	Quality	Client Jitter	Client RTT	Server Jitter	Server RTT
43.134.10.37	13.4.1.3	2016-04-02 20:20					
94.16.190.189	13.4.1.3	2016-04-02 19:52		16540 	1880 	1 	2
101.197.60.137	12.2.4.1	2016-04-02 20:38		11500 	3160 	2 	2
23.136.101.253	12.2.4.1	2016-04-02 20:40		4900 	2160 	2 	2
33.224.94.164	14.3.1.2	2016-04-02 20:15		11440 	3860 	2 	1

System Center Operations Manager

System Center Operations Manager is a monitoring product from Microsoft and is part of the System Center suite. Operations Manager is primarily an agent based monitoring tool but it also supports networking devices using SNMP.

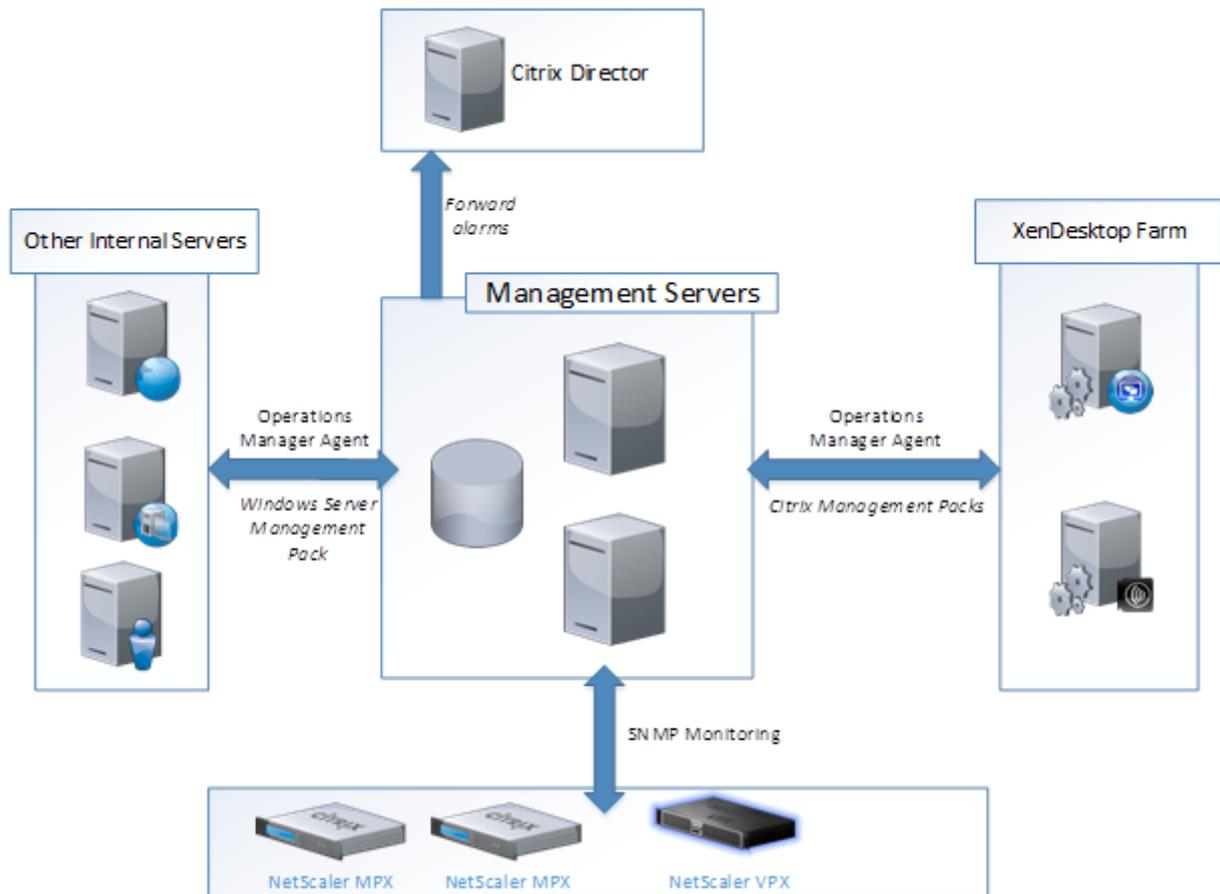
The benefit with Operations Manager is that it can monitor all pieces of the infrastructure, from the hardware level up to the particular application, and it can do end-to-end monitoring against web services for instance.

It has quite extensive monitoring capabilities against Windows Server and has different monitoring views/alerts for different Windows server components such as Active Directory, File Server, IIS and so on. This is done by installing agents on each server we want to monitor and we have to import something called management pack for those features we want to monitor as well. Management Packs contains the logic to do monitoring for a particular component such as Active Directory or Microsoft SQL. So for instance if I have an Operations Manager agent installed on a Active Directory server I will not get any Active Directory based monitoring unless I import an Active Directory management pack.

Now with 7.8 platinum license customers will also get access to Citrix management packs, which allows us to monitor our Citrix infrastructure (Storefront, delivery controllers, PVS and so on) directly in System Center. To monitor all Windows based Citrix components we must have an agent installed on each Citrix Server. All the data gathered by the agents are forwarded to a management server, where the data is aggregated into a SQL database.

We can also use System Center to monitor NetScaler appliances by configuring SNMP and using a Citrix management pack for NetScaler which will give System Center the knowledge about what information it needs to gather from the SNMP objects.

Operations Manager can also be integrated with Citrix Director to display alerts related to our Citrix infrastructure. So an example overview can be shown in the screenshot below



NOTE: Operations Manager will not display Appflow data, so it is not an alternative to Insight Center or Goliath IT analytics. It is more of a pure monitoring tool for system health, while Goliath and Insight is more to display end user connections and display statistics around those.