Implementing Client Certificate Authentication for ADFS Proxy on NetScaler

Solution Guide

This guide describes the implementation of client certificate based authentication for the ADFS Proxy solution on NetScaler.

Citrix NetScaler is a world-class product with the proven ability to load balance, accelerate, optimize, and secure enterprise applications.

The ADFS Proxy with client certificate authentication solution allows enterprises to utilize hardware and software based tokens with client certificates for authentication of users. This solution can facilitate secure, multi-factor authentication.

Introduction

One of the biggest concerns of any IT organization today is security. Customers are looking to securely provide network resources to their user community, regardless of the physical location of the user (on-premises, off-premises). As cloud computing offerings such as SaaS become more and more prevalent, organizations across all verticals including retail, healthcare, manufacturing, public sector & government and others are looking at these solutions to provide an economies of scale to grow their business functions and capabilities while adhering to their enterprise security policies.

Solution Benefits

- Secure appliance to mitigate malicious attacks
- NetScaler does not need to be domain joined
- Rewrite/URL Transformation for secure redirection
- SSL Offload and Acceleration
- FIPS Benefits (Secure Keys)
- Multi-Factor Authentication
- Centralized user and traffic visibility

Office 365 (O365) is becoming one of the fastest growing customer adoption cloud based solutions. As organizations look to move to O365 for mail, collaboration, voice & video capabilities they want to extend their security posture without negatively affecting the user experience. The NetScaler platform is the ideal solution for customers to protect their investment on premises and in the cloud. The features and capabilities of the NetScaler to provide federated identity with SAML and ADFS Proxy, enable Single Sign-On (SSO) with our multi-factor authentication element N-Factor allows it to seamlessly integrate into a customer's identity management and authentication solution. This enables IT organizations to provide the same user experience across all entry points. The SSL Offload capability allows the solution to decrypt encrypted traffic and provide additional security inspection (then re-encrypt if desired). It can also give customers the flexibility to mask and obfuscate their clients URL/URI with URL Rewrite and Transformation capabilities. The NetScaler solution altogether will provide customers a secure and seamless user experience to get access to both their on-premises and cloud resources.

The ADFS Proxy capabilities of NetScaler make it an effective part of the enterprise network for securing the internal ADFS environment by preventing it from being externally exposed. However, it is also necessary that users are still able to access authenticated applications transparently, without being made aware of the presence of the proxy. Preventing unauthorized access to the ADFS server (for users without valid credentials) and securing the proxy server with pre-authentication and other security-oriented capabilities is also an important use case. This solution guide addresses such use cases for enterprises requiring secure ADFS proxy deployments.

Configuration

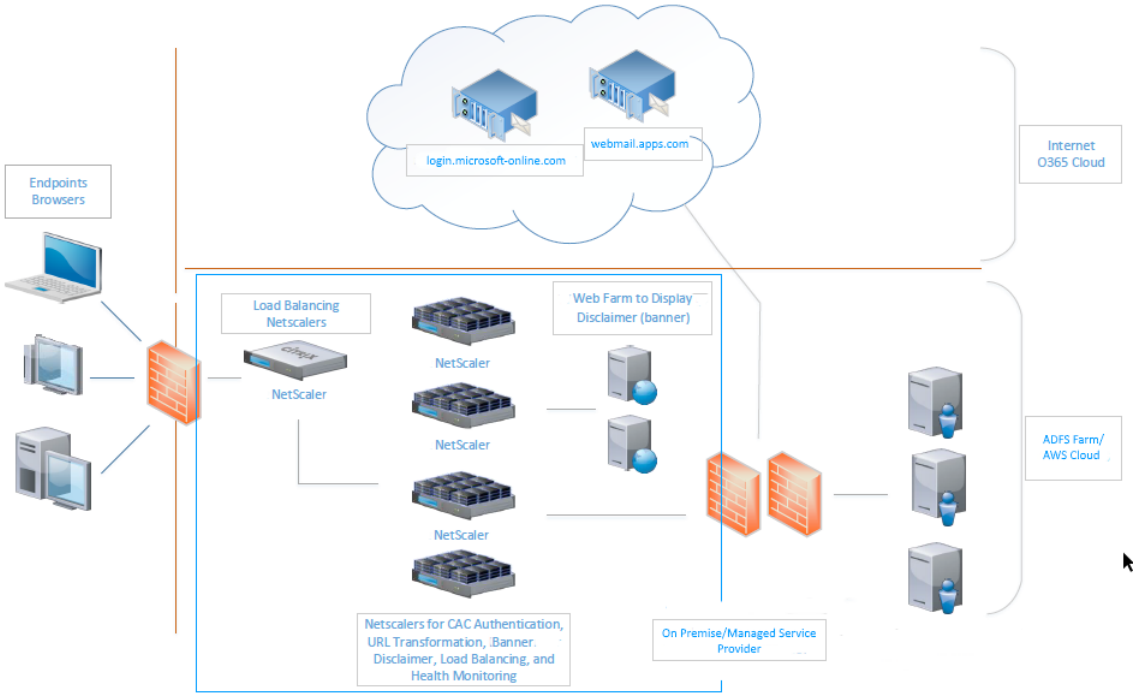
The table below lists the minimum required software versions for this integration to work successfully. The integration process should also work with higher versions of the same.

Recommended Product Versions

Product	Version
NetScaler	11.1 (Enterprise/Platinum License)

Solution Description

The following instructions assume that ADFS server side configuration has been completed. Please note that on the NetScaler, SNI bindings should be disabled for now on the service side, as we are currently testing the SNI binding functionality with ADFS 3.0.



Part 1: Configure NetScaler

Setup default NetScaler settings, and enable at least the following features – Load Balancing, Content Switching, AAA and Rewrite:

```
enable ns feature WL SP LB CS CMP SSL CF IC SSLVPN AAA REWRITE
enable ns mode FR L3 CKA Edge USNIP SRADV PMTUD
set system parameter -natPcbForceFlushLimit 4294967295 -doppler ENABLED
set system user nsroot 1fffa852a9aed1dn34k1jlk3739648d125df6b5e609 -encrypted
-timeout 36000
set rsskeytype -rsstype ASYMMETRIC
set lacp -sysPriority 32768 -mac f2:94:49:55:c4:b2
set interface 1/1 -throughput 0 -bandwidthHigh 0 -bandwidthNormal 0 -intftype
"Xen Virtual" -ifnum 1/1
set interface 1/2 -throughput 0 -bandwidthHigh 0 -bandwidthNormal 0 -intftype
"Xen Virtual" -ifnum 1/2
set interface LO/1 -haMonitor OFF -haHeartbeat OFF -throughput 0 -bandwidth-
High 0 -bandwidthNormal 0 -intftype Loopback -ifnum LO/1
add ns trafficDomain 4094
add ns ip6 fe80::3c7e:7dff:fe40:d2d2/64 -scope link-local -type NSIP -vlan 1
-vServer DISABLED -mgmtAccess ENABLED -dynamicRouting ENABLED
add ns ip 10.105.157.181 255.255.255.0 -vServer DISABLED
add ns ip 192.168.1.3 255.255.255.0 -vServer DISABLED -mgmtAccess ENABLED
```

(This configuration uses sample IP addresses, replace the same with the relevant ones for your environment)

Add the KCD account used for backend Kerberos:

```
add aaa kcdAccount kcd _ adfs _ proxy -realmStr TESTREALM.COM -delegatedUser
administrator -kcdPassword fa0fc1ef5209621785252dajd1237cfb4564b50d3546999f-
52f66385a94 -encrypted -encryptmethod ENCMTHD _ 3
```

Add the ADFS server and service:

```
add server adfs _ xenapp 10.105.157.131

add service adfs _ https _ xenapp adfs _ xenapp SSL 443 -gslb NONE -maxClient 0
-maxReq 0 -cip ENABLED X-MS-Forwarded-Client-IP -usip NO -useproxyport YES -sp
ON -cltTimeout 180 -svrTimeout 360 -CKA NO -TCPB NO -CMP YES
```

Add the required SSL certificates (in this setup, all certificates are created using an internal CA):

```
add ssl certKey ns-server-certificate -cert ns-server.cert -key ns-server.key
add ssl certKey adfs _ test _ cac -cert adfs _ test _ cac.cer -key adfs _ test _
cac.key
add ssl certKey wildcard _ TESTREALM -cert wildcard _ TESTREALM.cer -key wild-
card _ TESTREALM.key
add ssl certKey test _ clientcert -cert test _ clientcert.cer -key test _ client-
cert
add ssl certKey nsroot -cert ns-root.cert -key ns-root.key
add ssl certKey startcom -cert startcom.cer -inform DER
add ssl certKey ns-root-ca -cert ns-root-ca.cer
add ssl certKey ns-client-cert -cert ns-client-cert.cer -key ns-client-cert.key
add ssl certKey adfscac -cert adfscac.cer -key adfscac.key
```

Create the authentication and traffic policy/actions required (LDAP, Client certificate and Negotiate (for Kerberos) defined here:

```
add authentication ldapAction ldap _ 1 -serverIP 10.105.157.116 -ldapBase
"CN=Users,DC=TESTREALM,DC=COM" -ldapBindDn "CN=Administrator,CN=Users,DC=
TESTREALM,DC=COM" -ldapBindDnPassword 4fcb95f983ceaaabfd1867280484f5de8ffe-
12197a3f5fbbff766bec645acdf0 -encrypted -encryptmethod ENCMTHD _ 3 -ldapLogin-
Name sAMAccountName
add authentication negotiateAction adfs _ proxy _ neg -domain TESTREALM.com
-domainUser administrator -domainUserPasswd 4606206b6e7ccac0be0425494f506d228c-
3fa3934c5e980c653eeb55208b4943 -encrypted -encryptmethod ENCMTHD _ 3 -NTLMPath
"http://10.105.157.131/integrated.html"
add authentication certAction cert _ profile _ test -twoFactor ON -userNameField
Subject:CN


add tm trafficAction prf _ sso _ to _ 401-adfs -SSO ON -persistentCookie OFF -Ini-
tiateLogout OFF -kcdAccount kcd _ adfs _ proxy
```

(continued on the next page)

```
add tm trafficAction prf _ sso _ to _ 401-adfs -SSO ON -persistentCookie OFF -Ini-
tiateLogout OFF -kcdAccount kcd _ adfs _ proxy

add authentication ldapPolicy ldap _ svc _ pol ns _ true ldap _ 1
add authentication negotiatePolicy adfs _ negotiate ns _ true adfs _ proxy _ neg
add authentication certPolicy cert _ test ns _ true cert _ profile _ test
add tm trafficPolicy pol _ sso _ to _ 401-adfs "http.REQ.URL.PATH.EQ(\"/adfs/ls/
wia\")" prf _ sso _ to _ 401-adfs
add authentication vserver ldap _ auth _ vs SSL 10.105.157.138 443 -Authentica-
tionDomain TESTREALM.com -td 4094 -maxLoginAttempts 255 -failedLoginTimeout 1
```

Add CS server and policies, and LB vservers

```
add cs vserver vip _ adfs _ xenapp SSL 10.105.157.165 443 -cltTimeout 180
add lb vserver vip _ adfs _ https _ xenapp SSL 0.0.0.0 0 -persistenceType NONE
-cltTimeout 180
add lb vserver vip _ adfs _ http HTTP 0.0.0.0 0 -persistenceType NONE -cltTimeout
180
add cs action adfs _ xenapp -targetLBVserver vip _ adfs _ http
add cs policy pol _ adfs _ proxynoauth -rule "http.REQ.URL.CONTAINS(\"/adfs/ser-
vices/trust\")|| http.REQ.URL.CONTAINS(\"/federationmetadata/2007-06/federation-
metadata.xml\")" -action act _ lb _ adfs _ proxynoauth
add cs policy ADFSProxy _ Passive -rule "http.REQ.URL.CONTAINS(\"/adfs/ls/wia\")
|| http.REQ.URL.CONTAINS(\"/adfs/ls/auth/integrated\")" -action ADFSProxy _ Passive
add cs policy adfs _ xenapp -rule "HTTP.REQ.HOSTNAME.SET _ TEXT _ MODE(IGNORECASE).
EQ(\"adfscac.TESTREALM.com\") && HTTP.REQ.URL.SET _ TEXT _ MODE(IGNORECASE).CON-
TAINS(\"/adfs\")" -action adfs _ xenapp
```

Add required rewrite actions and the URL Transformation policy

```
add rewrite action rewrite _ adfs _ proxy replace http.REQ.URL "\"/adfs/services/
trust/proxymex\""
add rewrite action act _ rw _ adfs _ mexrequest replace http.REQ.URL "\"/adfs/ser-
vices/trust/proxymex\""
add rewrite action rewrite _ adfs _ ProxyHeader _ xenapp insert _ http _ header X-
MS-Proxy "\"NETSCALER\""
add rewrite action rewrite _ adfs _ Mex _ xenapp replace HTTP.REQ.URL.PATH _ AND _
QUERY "\"/adfs/services/trust/proxymex\" +
HTTP.REQ.URL.SET _ TEXT _ MODE(IGNORECASE).PATH _ AND _ QUERY.STRIP _ START _
CHARS(\"/adfs/services/trust/mex\").HTTP _ URL _ SAFE"
add rewrite policy rewrite _ adfs _ proxy _ pol "http.REQ.URL.CONTAINS(\"/adfs/ser-
vices/trust\") && (!HTTP.REQ.URL.CONTAINS(\"/trust/proxymex\"))" rewrite _ adfs _
proxy
```

(continued on the next page)

```
add rewrite policy pol _ rw _ adfs _ mexrequest "http.REQ.URL.CONTAINS(\"/adfs/
services/trust/mex\")" act _ rw _ adfs _ mexrequest
add rewrite policy rw _ pol _ adfs _ ProxyHeader _ xenapp "http.REQ.URL.TO _ LOWER.
STARTSWITH(\"/adfs\")" rewrite _ adfs _ ProxyHeader _ xenapp
add rewrite policy rw _ pol _ adfs _ Mex _ xenapp "http.REQ.URL.TO _ LOWER.
STARTSWITH(\"/adfs/services/trust/mex\")" rewrite _ adfs _ Mex _ xenapp
add rewrite policy Replace _ URL _ Test "http.REQ.HOSTNAME.CONTAINS(\"adfscac\")"
Request _ Replace _ Test
add rewrite policylabel test _ http _ response http _ res
add rewrite policylabel response _ rewrite url

add transform profile Rewrite _ Test
add transform action Rewrite _ Action Rewrite _ Test 100
set transform action Rewrite _ Action -priority 100 -reqUrlFrom adfsfake.TEST-
REALM.com -reqUrlInto adfscac.TESTREALM.com -resUrlFrom adfscac.TESTREALM.com
-resUrlInto adfsfake.TESTREALM.com
add transform policy rewrite _ response true Rewrite _ Test
```

Bind policies and services to LB vservers

```
bind lb vserver proxy _ adfs _ server adfs _ https _ xenapp
bind lb vserver vip _ adfs _ https _ xenapp adfs _ https _ xenapp
bind lb vserver vip _ adfs _ http adfs _ https _ xenapp
bind lb vserver proxy _ adfs _ server -policyName rewrite _ adfs _ proxy _ pol
-priority 100 -gotoPriorityExpression END -type REQUEST
bind lb vserver vs _ https _ adfsproxy _ noauth -policyName pol _ rw _ adfs _
mexrequest -priority 100 -gotoPriorityExpression END -type REQUEST
bind lb vserver vip _ adfs _ https _ xenapp -policyName rw _ pol _ adfs _ Proxy-
Header _ xenapp -priority 100 -gotoPriorityExpression NEXT -type REQUEST
bind lb vserver vip _ adfs _ https _ xenapp -policyName rw _ pol _ adfs _ Mex _ xe-
napp -priority 110 -gotoPriorityExpression END -type REQUEST
bind lb vserver vip _ adfs _ http -policyName rw _ pol _ adfs _ ProxyHeader _ xe-
napp -priority 100 -gotoPriorityExpression END -type REQUEST
bind lb vserver vip _ adfs _ http -policyName rw _ pol _ adfs _ Mex _ xenapp -pri-
ority 110 -gotoPriorityExpression END -type REQUEST
bind lb vserver vs _ https _ adfsproxy -policyName pol _ sso _ to _ 401-adfs -pri-
ority 100 -gotoPriorityExpression END -type REQUEST

bind cs vserver vip _ adfs _ xenapp -policyName rewrite _ response -priority 100
-gotoPriorityExpression END -type REQUEST
bind cs vserver vip _ adfs _ xenapp -policyName adfs _ xenapp -targetLBVserver
vip _ adfs _ https _ xenapp -priority 70
bind cs vserver vip _ adfs _ xenapp -lbvserver vip _ adfs _ http
```

Add SSL certificates and set SSL parameters, add DNS nameservers and LB monitor for the ADFS server (accesses federation metadata)

```
add dns nameServer 192.168.1.15
add dns nameServer 10.105.157.14
add dns nameServer 10.105.157.116 -dnsProfileName default-dns-profile

add lb monitor mon _ adfs _ https _ xenapp HTTP-ECV -customHeaders "host: adfs-
cac.TESTREALM.com\r\n" -send "GET /federationmetadata/2007-06/federationmetada-
ta.xml" -recv "adfstest.TESTREALM.com/adfs/services/trust" -LRTM ENABLED -se-
cure YES

add route 0.0.0.0 0.0.0.0 10.105.157.1
set ssl service adfs _ https _ xenapp -tls11 DISABLED -tls12 DISABLED
set ssl vserver proxy _ adfs _ server -cipherRedirect ENABLED -tls11 DISABLED
-tls12 DISABLED
set ssl vserver vip _ adfs _ https _ xenapp -clientAuth ENABLED -clientCert Man-
datory -tls11 DISABLED -tls12 DISABLED
set ssl vserver ldap _ auth _ vs -tls11 DISABLED -tls12 DISABLED
set ssl vserver vip _ adfs _ xenapp -clientAuth ENABLED -clientCert Mandatory
-tls11 DISABLED -tls12 DISABLED

add tm sessionPolicy adfs _ proxy _ sess ns _ true adfs _ sess _ prof
```

Binding of Certificates and SSL Settings

```
bind authentication vserver ldap _ auth _ vs -policy cert _ test -priority 100

bind ssl vserver proxy _ adfs _ server -cipherName ALL
bind ssl vserver proxy _ adfs _ server -cipherName DEFAULT
bind ssl vserver vip _ adfs _ xenapp -cipherName DEFAULT
bind ssl vserver vip _ adfs _ xenapp -cipherName ALL
bind ssl vserver adfs _ proxy _ activeserver -certkeyName exch2016
bind ssl vserver proxy _ adfs _ server -certkeyName wildcard _ TESTREALM
bind ssl vserver vs _ https _ adfsproxy -certkeyName exch2016
bind ssl vserver vs _ https _ adfsproxy _ noauth -certkeyName exch2016
```

```
bind ssl vserver ADFSProxy _ Passive -certkeyName exch2016
bind ssl vserver vip _ adfs _ https _ xenapp -certkeyName adfscac
bind ssl vserver vip _ adfs _ https _ xenapp -certkeyName ns-root-ca -CA -ocsp-
Check Optional
bind ssl vserver ldap _ auth _ vs -certkeyName exch2016
bind ssl vserver ldap _ auth _ vs -certkeyName nsroot -CA -ocspCheck Optional
bind ssl vserver adfs _ proxy _ aaa -certkeyName exch2016
bind ssl vserver adfs _ proxy _ passive _ aaa -certkeyName exch2016
bind ssl vserver vs _ aaa _ adfsproxy -certkeyName exch2016
bind ssl vserver cs _ exch2016 -certkeyName exch2016
bind ssl vserver cs _ adfs _ proxy -certkeyName adfs _ test _ cac
bind ssl vserver vs _ https _ adfs -certkeyName exch2016
bind ssl vserver vip _ adfs _ xenapp -certkeyName adfscac
bind ssl vserver vip _ adfs _ xenapp -certkeyName ns-root-ca -CA -skipCAName
-ocspCheck Optional
bind ssl vserver adfs _ proxy _ activeserver -eccCurveName P _ 256
bind ssl vserver adfs _ proxy _ activeserver -eccCurveName P _ 384
bind ssl vserver adfs _ proxy _ activeserver -eccCurveName P _ 224
bind ssl vserver adfs _ proxy _ activeserver -eccCurveName P _ 521
bind ssl vserver proxy _ adfs _ server -eccCurveName P _ 256
bind ssl vserver proxy _ adfs _ server -eccCurveName P _ 384
bind ssl vserver proxy _ adfs _ server -eccCurveName P _ 224
bind ssl vserver proxy _ adfs _ server -eccCurveName P _ 521
bind ssl vserver vs _ https _ adfsproxy -eccCurveName P _ 256
bind ssl vserver vs _ https _ adfsproxy -eccCurveName P _ 384
bind ssl vserver vs _ https _ adfsproxy -eccCurveName P _ 224
bind ssl vserver vs _ https _ adfsproxy -eccCurveName P _ 521
bind ssl vserver vs _ https _ adfsproxy _ noauth -eccCurveName P _ 256
bind ssl vserver vs _ https _ adfsproxy _ noauth -eccCurveName P _ 384
bind ssl vserver vs _ https _ adfsproxy _ noauth -eccCurveName P _ 224
bind ssl vserver vs _ https _ adfsproxy _ noauth -eccCurveName P _ 521
bind ssl vserver ADFSProxy _ Passive -eccCurveName P _ 256
bind ssl vserver ADFSProxy _ Passive -eccCurveName P _ 384
bind ssl vserver ADFSProxy _ Passive -eccCurveName P _ 224
bind ssl vserver ADFSProxy _ Passive -eccCurveName P _ 521
bind ssl vserver vip _ adfs _ https _ xenapp -eccCurveName P _ 256
bind ssl vserver vip _ adfs _ https _ xenapp -eccCurveName P _ 384
bind ssl vserver vip _ adfs _ https _ xenapp -eccCurveName P _ 224
bind ssl vserver vip _ adfs _ https _ xenapp -eccCurveName P _ 521
bind ssl vserver ldap _ auth _ vs -eccCurveName P _ 256
bind ssl vserver ldap _ auth _ vs -eccCurveName P _ 384
bind ssl vserver ldap _ auth _ vs -eccCurveName P _ 224
bind ssl vserver ldap _ auth _ vs -eccCurveName P _ 521
bind ssl vserver adfs _ proxy _ aaa -eccCurveName P _ 256
bind ssl vserver adfs _ proxy _ aaa -eccCurveName P _ 384
bind ssl vserver adfs _ proxy _ aaa -eccCurveName P _ 224
bind ssl vserver adfs _ proxy _ aaa -eccCurveName P _ 521
```

```
bind ssl vserver adfs_proxy_passive_aaa -eccCurveName P_256
bind ssl vserver adfs_proxy_passive_aaa -eccCurveName P_384
bind ssl vserver adfs_proxy_passive_aaa -eccCurveName P_224
bind ssl vserver adfs_proxy_passive_aaa -eccCurveName P_521
bind ssl vserver vs_aaa_adfsproxy -eccCurveName P_256
bind ssl vserver vs_aaa_adfsproxy -eccCurveName P_384
bind ssl vserver vs_aaa_adfsproxy -eccCurveName P_224
bind ssl vserver vs_aaa_adfsproxy -eccCurveName P_521
bind ssl vserver cs_adfs_proxy -eccCurveName P_256
bind ssl vserver cs_adfs_proxy -eccCurveName P_384
bind ssl vserver cs_adfs_proxy -eccCurveName P_224
bind ssl vserver cs_adfs_proxy -eccCurveName P_521
bind ssl vserver vs_https_adfs -eccCurveName P_256
bind ssl vserver vs_https_adfs -eccCurveName P_384
bind ssl vserver vs_https_adfs -eccCurveName P_224
bind ssl vserver vs_https_adfs -eccCurveName P_521
bind ssl vserver vip_adfs_xenapp -eccCurveName P_256
bind ssl vserver vip_adfs_xenapp -eccCurveName P_384
bind ssl vserver vip_adfs_xenapp -eccCurveName P_224
bind ssl vserver vip_adfs_xenapp -eccCurveName P_521
```

This completes the configuration for the ADFS Client Certificate solution on NetScaler.

Conclusion

NetScaler enables custom, secure ADFS deployments with its authentication and authorization capabilities. With NetScaler, enterprises can enable a host of additional capabilities including but not limited to authentication offload, end point analysis checks, selective server access, URL rewrites, compression, caching, front end optimizations and much more.

To learn more about how NetScaler can bring these benefits to address this and other enterprise requirements, please visit http://www.citrix.com.

CITRIX®

Enterprise Sales
North America | 800-424-8749
Worldwide | +1 408-790-8000

Locations
Corporate Headquarters | 851 Cypress Creek Road Fort Lauderdale, FL 33309 United States
Silicon Valley | 4988 Great America Parkway Santa Clara, CA 95054 United States

Add the KCD account used for backend Kerberos:

```
add aaa kcdAccount kcd _ adfs _ proxy -realmStr TESTREALM.COM -delegatedUser
administrator -kcdPassword fa0fc1ef5209621785252dajd1237cfb4564b50d3546999f-
52f66385a94 -encrypted -encryptmethod ENCMTHD _ 3
```

Add the ADFS server and service:

```
add server adfs _ xenapp 10.105.157.131

add service adfs _ https _ xenapp adfs _ xenapp SSL 443 -gslb NONE -maxClient 0
-maxReq 0 -cip ENABLED X-MS-Forwarded-Client-IP -usip NO -useproxyport YES -sp
ON -cltTimeout 180 -svrTimeout 360 -CKA NO -TCPB NO -CMP YES
```

Add the required SSL certificates (in this setup, all certificates are created using an internal CA):

```
add ssl certKey ns-server-certificate -cert ns-server.cert -key ns-server.key
add ssl certKey adfs _ test _ cac -cert adfs _ test _ cac.cer -key adfs _ test _
cac.key
add ssl certKey wildcard _ TESTREALM -cert wildcard _ TESTREALM.cer -key wild-
card _ TESTREALM.key
add ssl certKey test _ clientcert -cert test _ clientcert.cer -key test _ client-
cert
add ssl certKey nsroot -cert ns-root.cert -key ns-root.key
add ssl certKey startcom -cert startcom.cer -inform DER
add ssl certKey ns-root-ca -cert ns-root-ca.cer
add ssl certKey ns-client-cert -cert ns-client-cert.cer -key ns-client-cert.key
add ssl certKey adfscac -cert adfscac.cer -key adfscac.key
```

Create the authentication and traffic policy/actions required (LDAP, Client certificate and Negotiate (for Kerberos) defined here:

```
add authentication ldapAction ldap _ 1 -serverIP 10.105.157.116 -ldapBase
"CN=Users,DC=TESTREALM,DC=COM" -ldapBindDn "CN=Administrator,CN=Users,DC=
TESTREALM,DC=COM" -ldapBindDnPassword 4fcb95f983ceaaabfd1867280484f5de8ffe-
12197a3f5fbbff766bec645acdf0 -encrypted -encryptmethod ENCMTHD _ 3 -ldapLogin-
Name sAMAccountName
add authentication negotiateAction adfs _ proxy _ neg -domain TESTREALM.com
-domainUser administrator -domainUserPasswd 4606206b6e7ccac0be0425494f506d228c-
3fa3934c5e980c653eeb55208b4943 -encrypted -encryptmethod ENCMTHD _ 3 -NTLMPath
"http://10.105.157.131/integrated.html"
add authentication certAction cert _ profile _ test -twoFactor ON -userNameField
Subject:CN


add tm trafficAction prf _ sso _ to _ 401-adfs -SSO ON -persistentCookie OFF -Ini-
tiateLogout OFF -kcdAccount kcd _ adfs _ proxy
```

(continued on the next page)