End-To-End Encryption with XenApp and XenDesktop

Security guidance for Citrix Deployments

CITRIX®

Table of Contents

Protecting sensitive data often requires end-to-end encryption, meaning that data is encrypted from the point of service to the final point of termination, with no intervening decryption. This is in response to trends that include regulatory compliance, privacy legislation, contractual policies and vulnerability mitigations. These trends mean that many organizations need urgently to migrate to the use of TLS 1.2, and to manage cipher suites closely.

This white paper describes when and where to implement encryption, how to select encryption protocol options, and explains where to find detailed configuration guidance for the components of Citrix XenApp, XenDesktop, and NetScaler Gateway. It also discusses encryption in the cloud and in the Internet of Things, and the future direction of cryptography.

Most organizations now do not attempt to retain full control over use of corporate networks, as these networks often extend beyond organizational and management boundaries. Instead, they segment networks according to use cases and security levels (for example, segmenting less-trusted wireless networks available to visitors from more-trusted datacenter networks). Meanwhile, organizations have also realized that the insider threat is greater than previously believed. These threats can be perpetuated by malicious insiders, honest mistakes, and misconfigurations. Staff who are tricked by spear-phishing attacks, and "trusted" network devices already compromised by external attacks clearly illustrate an insider threat that simulates—but does not require—a rogue individual. Attackers are motivated to attack corporate networks not just to obtain corporate assets, but also personal data that can be used against individuals for identity theft or spear phishing.

SSL and TLS

The TLS (Transport Layer Security) protocol has superseded SSL. Although many products support both SSL and TLS, and the term "SSL" is often used to describe both, the difference between SSL and TLS is crucial.

Use TLS. SSL is no longer secure.

The combination of disruptive forces—motivated insider threats on less trusted networks—means that data must be protected by end-to-end encryption not just when crossing the Internet, but also when traversing corporate networks. This is reflected in recent regulatory requirements, including updates to the PCI Data Security Standard. With end-to-end encryption, data is protected throughout the data lifecycle, including data at rest, data in transit, and data in use.

However, blanket use of end-to-end encryption is not always an effective approach, due in part to needs for content inspections, logging, and traffic management. This white paper explains a prescribed use of end-to-end encryption with XenApp and XenDesktop, with specific details for configuring the Transport Layer Security (TLS) protocol.

This white paper builds on the guidance in the white paper "Getting Started with Citrix XenApp and XenDesktop Security" white paper, including the representative deployment described there. This paper is designed to meet the needs of security specialists, systems integrators, and consultants designing, deploying, and securing Citrix deployments.

Data in transit – encryption and the TLS protocol

The TLS protocol has evolved steadily since its origins in the Secure Sockets Layer (SSL) protocol in 1995. (TLS and SSL are still informally known as SSL.)

This section describes in detail the cryptography used by TLS, the options available, and how to select between them. This detail is important, because organizations may need to select less-secure options for compatibility with earlier operating systems, or temporarily while upgrading components.

Successive versions of the protocol:

- Correct security weaknesses in earlier versions
- Add support for new cryptography
- Add new protocol features

Cryptography in the TLS protocol

Cryptography in the TLS protocol is selected by a TLS cipher suite, which is negotiated between the client and server. This defines the cryptographic algorithms that are used for the connection.

Some versions of the TLS protocol permit cryptographic algorithms that are now considered weak, and should not be used. (However, not all usages of these algorithms within the TLS protocol are insecure—it depends on the context.) Insecure usage can be prevented by disabling the corresponding cipher suites. Many Citrix products automatically prevent use of insecure cipher suites, or disable them by default.

Your organizational policy may require specific cipher suites, and this may be determined by regulations, as explained later.

Cipher suites are named in the form TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384. This can be interpreted as follows:

- TLS is the protocol (Transport Layer Security)
- ECDHE_RSA is the key exchange algorithm (Elliptic Curve Diffie-Hellman)
- AES_256_CBC is the cipher (Advanced Encryption Standard, Cipher Block Chaining)
- SHA384 is the (MAC) message authentication code (Secure Hash Algorithm)

Examining the key exchange algorithm, ECDHE indicates that this cipher suite offers forward security. RSA indicates that a RSA digital certificate must be used.

Examining the cipher, AES_256_CBC indicates that this cipher suite uses a 256-bit key in CBC mode.

Examining the MAC, SHA384 indicates that this cipher suite uses the HMAC-SHA386 algorithm.

The cipher suite does not identify the version of the TLS protocol and many cipher suites are common to different TLS versions.

Note: *The naming scheme above is the one from the TLS standards. Some implementations, including OpenSSL and Citrix NetScaler, use a slightly different naming scheme for historical reasons. (For example, TLS1.2-ECDHE-RSA-AES-256-CBC-SHA384 corresponds to TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384.)*

HMAC functions

In cipher suites, MD5 and SHA1 algorithms are used as HMAC-MD5 and HMAC-SHA1 message authentication codes. These HMAC constructions are not considered weak.

However, some regulations do not permit the use of MD5 for any purpose. Therefore it is recommended that MD5 cipher suites are avoided. SHA1 cipher suites are not affected.

(MD5 and SHA1 used for signatures in digital certificates are weak.)

Strong and weak cipher suites

In this context, a weak cipher suite is one that can be attacked successfully now or projected in the next few years. (An attack may be difficult, but is at least possible.) Advances in technology, tools and techniques may weaken ciphers well before their initially projected lifespan and the known strength of ciphers should be periodically verified through NIST and other trusted sources.

Cipher suites containing the following algorithms are generally considered weak:

- DES (although 3DES—also known as TripleDES or TDEA—is not generally considered weak)
- RC2
- RC4

Additionally, the so-called 'export' or 'step-down' cipher suites are weak. These cipher suites limit the length of the signing key to 512 bits, which can be broken by brute force. These weak 'export' cipher suites were devised to satisfy export considerations that have not applied for many years. Strong cipher suites can and should be used instead.

Hashing algorithms including the SHA1 and MD5 are also considered weak for signatures in digital certificates, with SHA-256 being specified as the minimum standard. The usage of previous algorithms is so weak that public certificate authorities will no longer issue certificates that use them. Digital certificates using MD5 or SHA1 should be replaced. Some platforms, including Microsoft Windows, are already preventing their use.

In summary, for TLS today, the following are considered weak:

Algorithm	In certificates	In ciphersuites
DES	–	Weak
RC2	–	Weak
RC4	–	Weak
3DES (TDEA)	–	Not weak
MD2	Weak	–
MD5	Weak	Not weak, but avoid
SHA1	Weak	Not weak

Preferred cipher suites

AES is a block cipher: every block cipher is used in a particular mode of operation. Three of these modes have been standardized within TLS, as part of the cipher definition:

- AES-CBC (Cipher Block Chaining)
- AES-CCM (Counter with Cipher Block Chaining-Message Authentication Code). This mode is rarely used.
- AES-GCM (Galois Counter Mode)

The CBC mode is more widely supported than GCM, including in TLS version 1.0 and version 1.1.

The GCM mode is often preferred to CBC mode, because:

- It is higher-performance
- It is resistant to side-channel attacks, specifically padding oracle attacks such as Lucky Thirteen. (However, such attacks against CBC mode can be mitigated in other ways.)
- It is resistant to adaptive plaintext attacks, specifically the BEAST (Browser Exploit Against SSL/TLS) attack. (Again, this attack against CBC mode is mitigated in version TLS 1.1 or in other ways.)

Some authorities prefer GCM: others nevertheless still prefer CBC.

Unnecessary cipher suites

You only need to enable cipher suites that match the type of digital certificate you have installed—that is, which match the key exchange algorithm. For example, assuming you are using an RSA certificate, disable cipher suites that use ECDSA or DSS.

There are also cipher suites that do not use digital certificates. These are intended for special purposes that are outside the scope of this white paper. The usage of these should be explicitly disabled.

There are also cipher suites that do not encrypt at all. These are also intended for special purposes that are outside the scope of this white paper. The usage of these should be explicitly disabled.

The TLS standards also reference the following algorithms which are not widely supported or used: IDEA, Camellia, ARIA, GOST 28147-89. The usage of these should be explicitly disabled, unless needed for an exceptional case.

Note: *This is not an exhaustive list of cipher suites to avoid.*

Because of the large number of cipher suites, many of which are weak or not useful, it is by far safest only to enable the cipher suites that are actually needed. For most deployments, a handful will suffice. Ensure that all clients, including browsers, support the restricted set of cipher suites and test for proper operation as a key component of any client-side upgrade planning process.

Planning for the future

TLS version 1.3 will introduce newer cryptography, which is explained below. TLS version 1.3 also removes many aging cipher suites. Minimizing cipher suite usage will also simplify your future upgrade to TLS version 1.3.

Versions of the protocol

Different versions of the protocol do not interoperate; so that an SSL version 3 client cannot communicate with a TLS version 1.2 server, for example. Clients and servers typically implement several versions of the protocol to avoid overall interoperability problems.

However, there is a risk in clients and servers that support more than one version of the protocol. Recently, a set of 'cross-protocol' attacks have been found. One of these affects a client which supports SSL version 2, even when it is using TLS version 1.2. It is therefore important to disable, both on the client side and server side, versions of SSL and TLS protocols that are not required. For example, if only TLS version 1.2 is required, configure both clients and servers to support TLS version 1.2 only. This type of precaution will still be needed in the future, as newer versions replace older versions. Consult documentation for the corresponding Citrix components.

SSL – the predecessor to TLS

Both SSL version 2 and version 3 have security weaknesses, and should no longer be used. Both are officially deprecated by the IETF. If you have systems and applications that still rely on SSL version 2 and version 3, plan their migration to TLS version 1.2 immediately.

At this time, some Citrix products continue to support SSL version 3 for existing deployments. Support for SSL version 3 is being phased out.

TLS 1.0

TLS 1.0 is widely supported, including in Citrix products. TLS 1.0 is vulnerable to the BEAST (Browser Exploit Against SSL/TLS) attack, but this attack can be mitigated independently of TLS. TLS 1.0 is not recommended for new deployments, and some regulations (for example, PCI DSS 3.2) do not permit TLS 1.0 for new deployments.

TLS 1.1

TLS 1.1 is less widely supported, because it was quickly succeeded by TLS 1.2. It does not support newer cryptography, so TLS 1.2 is usually preferred. TLS 1.1 and later versions are not vulnerable to the BEAST attack.

TLS 1.2

TLS 1.2 is the preferred version for current deployments. It added cipher suites for newer cryptography (in particular, for SHA256 and SHA384 and for GCM mode). It is recommended or required by various regulations.

TLS 1.3 – the future

TLS 1.3 is currently under development. Unlike earlier versions of TLS, it removes many algorithms and little-used protocol features in the interest of security. It also introduces new cipher suites for extra cryptographic algorithms that are not supported in TLS 1.2 (including new elliptic curves ED25519 and ED488).

TLS and DTLS

Network protocols are of two kinds: connection-oriented (based on TCP) and connectionless (based on UDP). TLS is used with TCP; DTLS is used with UDP.

UDP-based protocols are often used for applications that are designed to cope with traffic loss, typically high-performance media streaming applications that deal with dropped frames but do not tolerate delays. DTLS allows for traffic loss, but TLS does not.

DTLS is a close derivative of TLS. It provides end-to-end encryption using the same digital certificates and cipher suites, but the DTLS protocol has a different handshake mechanism. Somewhat confusingly, DTLS 1.0 is equivalent to TLS 1.1, and DTLS 1.2 is equivalent to TLS 1.2.

DTLS is not as widely deployed and supported as TLS. This is partly because UDP-based protocols are more difficult to cross firewalls, and partly because UDP-based protocols are less widely used.

Organizational policies may not cover the need for DTLS, or may not distinguish between TLS and DTLS.

Citrix has supported TLS for many years, and has introduced DTLS support to Citrix Receiver and NetScaler Gateway to support Citrix FrameHawk technology and UDP audio. Contact your Citrix representative if you have specific needs for DTLS support.

Protecting encryption keys

Encryption is only as strong as the keys it uses. In TLS, the main keys to be protected are the private keys for digital certificates.

Private keys should never leave the machine. Ideally, they should be even more tightly confined, with keys generated and stored within security hardware. For a client certificate, this is typically within a smartcard. For a server certificate, this is within a hardware security module (HSM). Consult NetScaler Gateway documentation for HSM options.

Private keys used for signing, as in TLS, should also never be imported or exported. However, there are reasons why this is sometimes done:

- To simplify certificate issuance: the private key is generated elsewhere and its digital certificate is issued; both are then imported to the machine—avoid doing this. Generate the private key on the machine itself.
- For backup purposes: the private key and its digital certificate are exported, so a replacement machine can be made available when needed- avoid doing this. Have a replacement machine ready with a separate private key and digital certificate ready.

- For replicas: the private key and its digital certificate are exported to all replica machines. This requires a special-purpose export mechanism so that the private key is only made available to replica machines and no others. Consult NetScaler Gateway documentation for high-availability options.
- For machines that share a certificate (which may be a wildcard certificate or not): the private key and its digital certificate are exported to all sharing machines. These machines may not be true replicas. Again, this should use a special-purpose export mechanism so that the private key is only made available to machines which match that certificate.
- For stateless virtual machines: these require a specific solution, as they cannot store a private key persistently because they reset automatically when turned off. See below for a solution for XenApp and XenDesktop virtual desktops.

Encryption policies and standards

Organizations need to select encryption policies and standards to match business needs. These policies and standards should then be enforced consistently, using appropriate technology.

Applying consistent policies

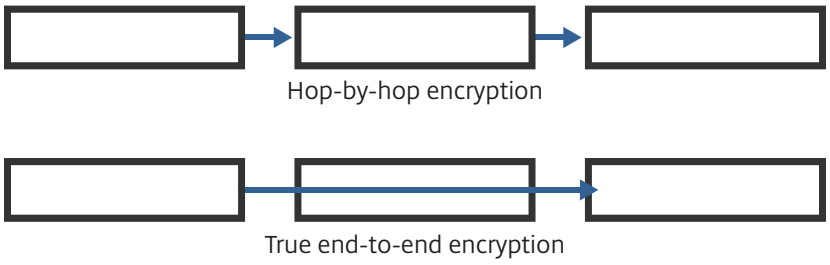
With end-to-end encryption, data is protected throughout the data lifecycle, including data at rest, data in transit, and data in use. Consistent protection is required throughout the data lifecycle; for example, to avoid strong protection for data at rest being weakened when in transit. This means applying consistent policies for encryption algorithms, encryption key lengths, and so on, wherever encryption is used.

This white paper focuses on encryption for data in transit, as supported by XenApp, XenDesktop, and NetScaler Gateway. For encryption of data at rest on mobile devices, consult Citrix XenMobile documentation.

Encryption for data in use is ultimately determined by application and data-level services (for example, a database application or a mobile container). Citrix ShareFile provides managed mobile container solutions, as part of the ShareFile app. Consult documentation specific to individual applications for detailed information on application-level encryption.

Data in transit: End-to-end and hop-by-hop encryption

For data in transit protection of virtual applications and desktops, there are two basic approaches; true end-to-end encryption, and hop-by-hop encryption.



With true end-to-end encryption, data is encrypted in transit from its ultimate source (for example, a client device) to its ultimate destination (for example, a database). This method is seen in encryption from point-of-swipe for credit card transactions. The advantage of true end-to-end encryption is that the data can be protected for confidentiality, integrity and availability everywhere. However, this is not always necessary or desirable. Connections often need to pass through firewalls, gateways, and proxy servers, which cannot inspect true end-to-end encrypted traffic. True end-to-end encryption also means that the client must know the true ultimate destination, which gateway proxy servers are intended to conceal.

Therefore, hop-by-hop encryption is often used. This is better suited to the TLS and IPSec protocols, which are generally deployed in hop-by-hop configurations. This means that data is protected when "on the wire"; however, with hop-by-hop encryption the data is not encrypted at the firewall, gateway, or proxy server itself.

Encryption policies do not usually distinguish between true end-to-end and hop-by-hop encryption. So, in the rest of this white paper, the term end-to-encryption covers both true end-to-end and hop-by-hop encryption. Implementation-specific details for systems and services intervening in end-to-end encryption will be supplemented with guidance.

Data classification and encryption

Your organization should have a data classification policy. This policy should take into account your organization's regulatory, governance and contractual commitments that drive the need for, and specification of, encryption.

Many organizations have three data classifications: public data, private data internal to the organization, and sensitive private data available only for specific groups of people. These classifications may be extended to cover customer information, and personally identifiable information in accordance with laws and regulations. The data classification policy should specify which data is to be kept confidential, as well as management needs for integrity and availability.

Public data need not be encrypted, by definition. Sensitive private data may well need to be encrypted. The data classification policy may not itself specify whether encryption is to be used; this may be detailed in other more specific organizational policies.

What to encrypt, and not to encrypt

Encryption should be used only when necessary, because:

- Encryption has a cost—for example, the operational cost in purchasing and managing digital certificates, and the performance overhead of encryption (however small)
- There are legal and regulatory restrictions on the use of encryption
- Encrypted data cannot easily be scanned, for example to check for viruses or undesired content

Not all data needs to be kept confidential (and therefore encrypted), whether at rest, in transit, or in use.

Your organization's data classification policy should be a starting point for determining the need for end-to-end encryption. However, some policies only cover application data (such as documents and databases), but not system data (such as credentials and keys). Credentials and private or secret keys must always be encrypted.

Administrative traffic (for example, between an administrative console and the server it controls) should always be encrypted. In some cases, traffic is automatically encrypted, and therefore does not require independent encryption. Where relevant, this is noted later in this white paper.

Regulated use of encryption

Over the years, encryption regulations have been simplified. Regulations still apply to national and international use, and to specific sectors.

National and international regulation

Encryption is subject to international regulation, for export. Some countries also regulate the import or use of encryption. This is separate from any industry–specific regulation in the financial services, telecommunications, or other sector.

Citrix publishes export information for Citrix products on the Citrix website (see References section of this paper for further details). For information about importing Citrix products please contact your Citrix representative.

Citrix cannot provide country-specific advice about the use of encryption. This includes deployments that span more than one country. Please contact your legal advisers for further guidance.

Sector-specific regulation

Organizations in the government, healthcare, energy, and financial services sectors are likely to be required to use encryption according to specific policies. These policies typically specify the use of particular encryption algorithms and key lengths. These implementation considerations are discussed in detail below, in the context of TLS.

Cryptographic validation – FIPS 140

Some sectors not only require use of particular cryptography, but also require that the implementation is independently validated. This is typical of government systems.

The U.S. Government requires that cryptography in Federal systems has been validated under the FIPS 140 validation program. Implementation of FIPS 140 is specified by levels that may include a hardware security module (HSM) for key protection. For details of cryptographic validation for XenApp and XenDesktop and its components, see the References section of this paper.

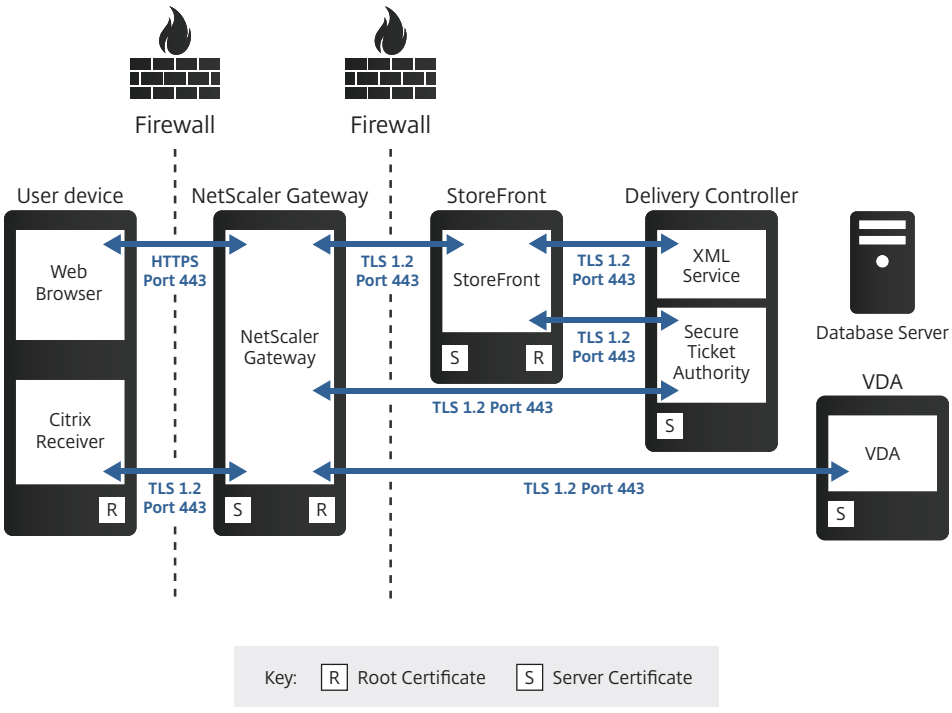
End-to-end encryption with XenApp and XenDesktop

This applies to XenApp and XenDesktop 7.6 and later.

Refer to "Payment Card Industry and Citrix XenApp and XenDesktop Deployment Scenarios" documentation to understand how these types of deployments help satisfy the Payment Card Industry Data Security Standard (PCI DSS).

Representative XenApp and XenDesktop configuration

As explained in "Getting Started with Citrix XenApp and XenDesktop Security" white paper, the configuration below provides remote access with end-to-encryption using TLS 1.2. This is the simplest layered configuration for remote access. The configuration shows which components are installed on different computers. The firewalls in this configuration should be configured to isolate the back-end components (StoreFront, Delivery Controller, and VDA) from other networks. (Refer to the "Getting Started with Citrix XenApp and XenDesktop Security" white paper for component requirements, including necessary versions of components.)



This white paper provides general guidance for TLS configuration for each of the link (hops) in this diagram. For step-by-step instructions, refer to product documentation for the version of the component involved.

Each link is administered differently, but has common guidance that covers:

- Preventing unencrypted communication
- Enforcing TLS 1.2
- Selecting ciphersuites
- Deploying digital certificates

Deploying digital certificates

Automate this process for all components if you can. You will need different procedures for Windows and non-Windows components, and for public and non-public certificates.

Do not use default certificates in a production deployment.

In some cases, configuration is recommended at both ends of the link, and this needs to be configured separately.

Similarly, where a component has more than one link (for example, StoreFront as shown above), in general these links need to be configured separately. These configuration steps can be combined, but for clarity are explained separately below.

Where possible, consider using Microsoft Group Policy to automate configuration across these components. Updates to Microsoft Windows (including as part of the automatic Windows Update process) have previously included changes to Microsoft-supported ciphersuites. Questions about these changes should be directed to Microsoft Corporation.

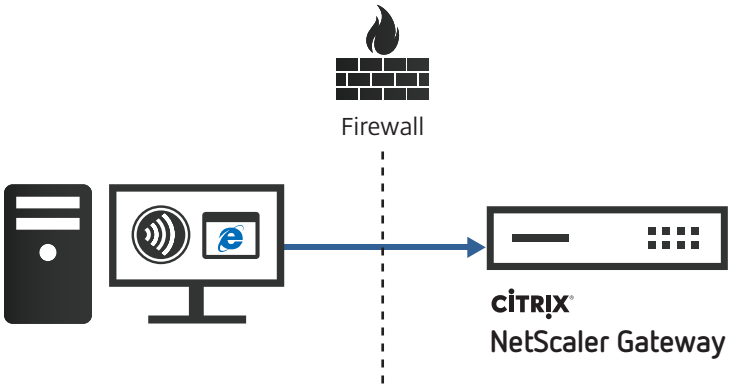
If a FIPS configuration is required, also refer to "Citrix XenApp 7.6 and XenDesktop 7.6 FIPS 140-2 Sample Deployments" documentation. This affects all components of the deployment.

Note: *Specific versions of components may be required.*

As shown in the diagram above, all TLS communication uses the standard TCP port 443. Refer to Citrix product documentation for use of different port numbers for TLS and other protocols; corresponding changes to network firewalls and operating system firewalls will be needed. This type of configuration is not generally recommended, but may be necessary if other applications or services are using the standard ports.

User device to NetScaler Gateway

This section describes TLS configuration for the link between the user device and NetScaler Gateway.



To prevent unencrypted communications, configure NetScaler Gateway to enforce TLS only, disabling all other protocols, such as telnet, FTP, and HTTP. Additionally, confirm that the external firewall has been configured so that NetScaler Gateway cannot be bypassed.

To enforce TLS 1.2, configure NetScaler Gateway vServer to enable only the TLS 1.2 protocol, and disable all others. Additionally, on each user device, configure Citrix Receiver, the web browser (if used) and the underlying operating system to enable only the TLS 1.2 protocol. Refer to Citrix Receiver documentation.

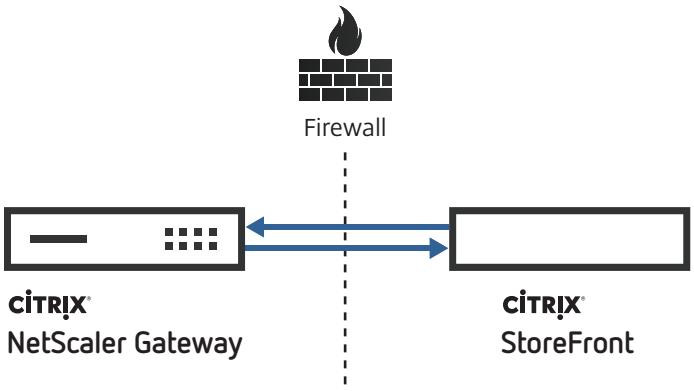
To select ciphersuites, configure NetScaler Gateway vServer to enable only the required ciphersuites, and disable all others. Additionally, on each user device, configure Citrix Receiver, the web browser (if used) and the underlying operating system to enable only the required ciphersuites. Refer to Citrix Receiver documentation.

To deploy digital certificates, refer to NetScaler product documentation. NetScaler Gateway is Internet-facing. Its server certificate should be obtained from a suitable public certification authority (CA), according to your organization's security policy.

NetScaler Gateway to StoreFront

This section describes TLS configuration for the link between NetScaler Gateway and StoreFront.

Note: *There are connections in both directions.*



Communication between NetScaler Gateway and StoreFront involves two TLS connections: one from NetScaler Gateway to StoreFront, and another from StoreFront to NetScaler Gateway. The connection from StoreFront to NetScaler is required for SmartAccess usage, and recommended for all configurations; it is configured via the Callback URL in the Authentication Settings of StoreFront.

To prevent unencrypted communications, configure Microsoft IIS on the StoreFront server to disable HTTP binding. Additionally, confirm that the internal firewall has been configured so that StoreFront cannot be bypassed.

To enforce TLS 1.2, at StoreFront (the TLS server) use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider.

Note: *This configuration will affect all other usage of Microsoft Schannel on this computer.*

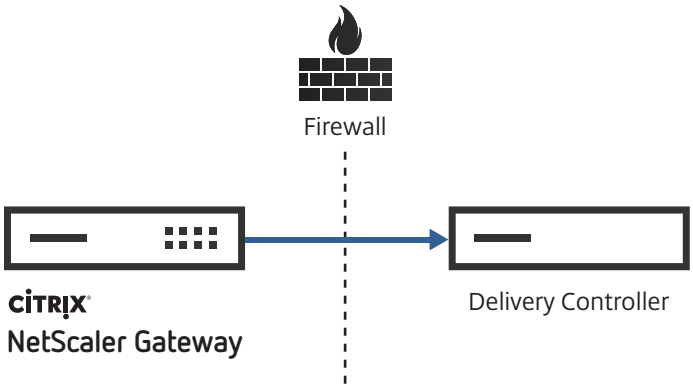
To select ciphersuites, at StoreFront (the TLS server) use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider.

Note: *This configuration will affect all other usage of Microsoft Schannel on this computer.*

To deploy digital certificates, follow Microsoft procedures to obtain, install, and configure a TLS server certificate on StoreFront according to your organization's security policy. Also install the corresponding root certificate on NetScaler Gateway. For SmartAccess usage (which requires configuring the Callback URL as explained above), install the root certificate corresponding to NetScaler Gateway on StoreFront.

NetScaler Gateway to Delivery Controller

This section describes TLS configuration for the link between NetScaler Gateway and Delivery Controller.



To prevent unencrypted communications, configure Delivery Controller to enforce use of HTTPS only. (This requires setting a registry value on the Delivery Controller.)

To enforce TLS 1.2, use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider on the Delivery Controller.

Note: *This configuration will affect all other usage of Microsoft Schannel on this computer.*

To select ciphersuites, use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider. To enforce selection of a particular ciphersuite, it may also be necessary to alter the priority order of ciphersuites in this configuration.

Note: *This configuration will affect all other usage of Microsoft Schannel on these computers.*

To deploy digital certificates, follow Microsoft procedures to obtain, install, and configure a TLS server certificate on the Delivery Controller according to your organization's security policy. Also install the corresponding root certificate on NetScaler Gateway.

Note: *Microsoft IIS can be installed on the Delivery Controller, but Delivery Controller does not require it.*

StoreFront to Delivery Controller

This section describes TLS configuration for the link between the StoreFront and Delivery Controller.

To prevent unencrypted communications, configure StoreFront and edit the Delivery Controller configuration to select Transport type HTTPS. Additionally, configure Delivery Controller to enforce use of HTTPS only. (This requires setting a registry value on the Delivery Controller.)

To enforce TLS 1.2, use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider. Apply this configuration both to StoreFront (the TLS client) and the Delivery Controller (the TLS server).

Note: *This configuration will affect all other usage of Microsoft Schannel on these computers.*

To select ciphersuites, use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider. Apply this configuration both to StoreFront and to the Delivery Controller. To enforce selection of a particular ciphersuite, it may also be necessary to alter the priority order of ciphersuites in this configuration.

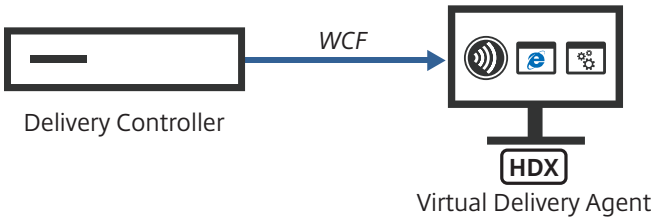
Note: *This configuration will affect all other usage of Microsoft Schannel on these computers.*

To deploy digital certificates, follow Microsoft procedures to obtain, install, and configure a TLS server certificate on the Delivery Controller according to your organization's security policy. Also install the corresponding root certificate on StoreFront.

Note: *Microsoft IIS can be installed on the Delivery Controller, but Delivery Controller does not require it.*

Delivery Controller to Virtual Delivery Agent (VDA)

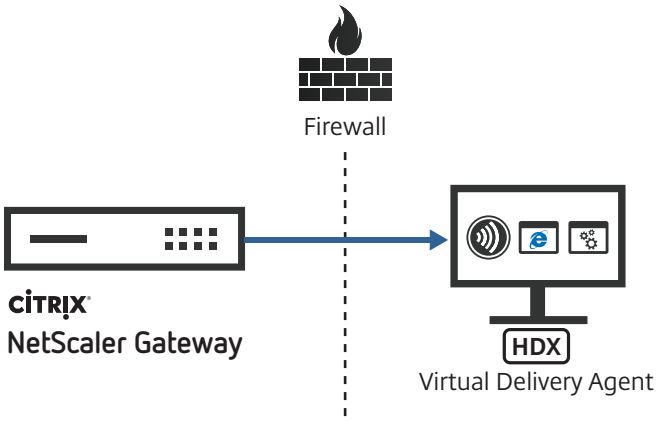
This section describes configuration for the link between Delivery Controller and VDA.



This connection is automatically encrypted using the WCF (Windows Communication Framework) protocol with Web Services Security Kerberos Token Profile 1.1. Therefore, separate configuration to use TLS 1.2 is not generally recommended, and is outside the scope of this white paper.

NetScaler Gateway to Virtual Delivery Agent (VDA)

This section describes TLS configuration for the link between NetScaler Gateway and VDA:



Special considerations apply to the VDA: If TLS is enabled for one VDA in a Delivery Group, it must be enabled for all VDAs in a Delivery Group, and so server certificates must be installed for all of them

Note: *The Linux VDA does not support TLS today.*

To prevent unencrypted communications, refer to XenApp and Desktop product documentation. A PowerShell script is provided that prevents unencrypted communications when TLS is enabled for a VDA.

To enforce TLS 1.2, refer to XenApp and XenDesktop product documentation. A PowerShell script is provided to select versions of the TLS protocol.

To select ciphersuites, refer to XenApp and XenDesktop product documentation. A PowerShell script is provided to select sets of ciphersuites.

To deploy digital certificates, refer to XenApp and XenDesktop product documentation. For the VDA, the typical approach depends on the machine catalog and the desktop experience, as follows:

Machine catalog	Desktop experience	Digital certificate	Install/update
Server OS		Shared	Master image
Desktop OS	Pooled random	Shared	Master image
Desktop OS	Pooled static	Shared	Master image
Desktop OS	Dedicated	Per machine	Auto-enrolment
Remote PC		Per machine	Auto-enrolment

In summary, for the VDA:

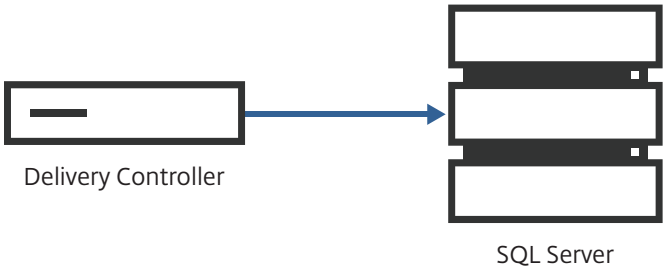
- **Non-persistent** – If you install the server certificate as part of a master image, the machines will need to share a certificate (likely a wildcard certificate). This applies to Server OS machine catalogs, and also to Desktop OS machine catalogs with the Desktop Experience being either Pooled random or Pooled static. Replace the certificate by updating the master image. (This is the stateless virtual machine case described above.) Any Machine Management method can be used.

- **Persistent** – If you install the server certificate via auto-enrolment to Active Directory Certificate Services (when the machine restarts), each machine can have its own certificate. This applies to Desktop OS machine catalogs with the Desktop Experience being Dedicated, and also to Remote PC machine catalogs. Use auto-reenrollment to Active Directory Certificate Services to replace the certificate when required. A PowerShell script is provided to simplify these processes at the VDA. Additionally, a scheduled task is required within the master image.

You can also install the certificate via auto-enrolment to Active Directory Certificate Services for Desktop OS machine catalogs with the Desktop Experience being either Pooled random or Pooled static. Each machine will then have its own certificate, avoiding the increased risk of exposure of a wildcard certificate. However, a new certificate will then be installed on every machine restart. If there is more than one server certificate present on the VDA, the PowerShell script mentioned above can select the appropriate certificate.

Delivery Controller to SQL Database

This section describes TLS configuration for the link between Delivery Controller and SQL Database.



To prevent unencrypted communications, configure Microsoft SQL Server to force encryption. (As explained in Citrix Knowledge Center article CTX1377556, it is also possible to force encryption at the Delivery Controller, or for each service. However, these alternatives have no advantage in production environments.)

To enforce TLS 1.2, it may be necessary to install a Microsoft update on Microsoft SQL Server. It is also necessary to configure registry settings on both Delivery Controller and database server, and other updates may also be required; refer to Microsoft article 3135244. Updates are available for SQL Server 2014 SP1, and SQL Server Express 2012 (which is supplied with Desktop Delivery Controller). Use SQL Server 2014 in a production environment. Microsoft recommends the use of TLS 1.2 in this configuration.

To select ciphersuites, use Microsoft Group Policy, PowerShell cmdlet, or registry configuration tools to configure the Microsoft Schannel provider. Apply this configuration both to the Delivery Controller and to the Microsoft SQL Server. To enforce selection of a particular ciphersuite, it may also be necessary to alter the priority order of ciphersuites in this configuration.

Note: *This configuration will affect all other usage of Microsoft Schannel on these computers.*

To deploy digital certificates, follow Microsoft procedures to obtain, install, and configure a TLS server certificate on the Microsoft SQL Server. Also install the corresponding root certificate on the Delivery Controller.

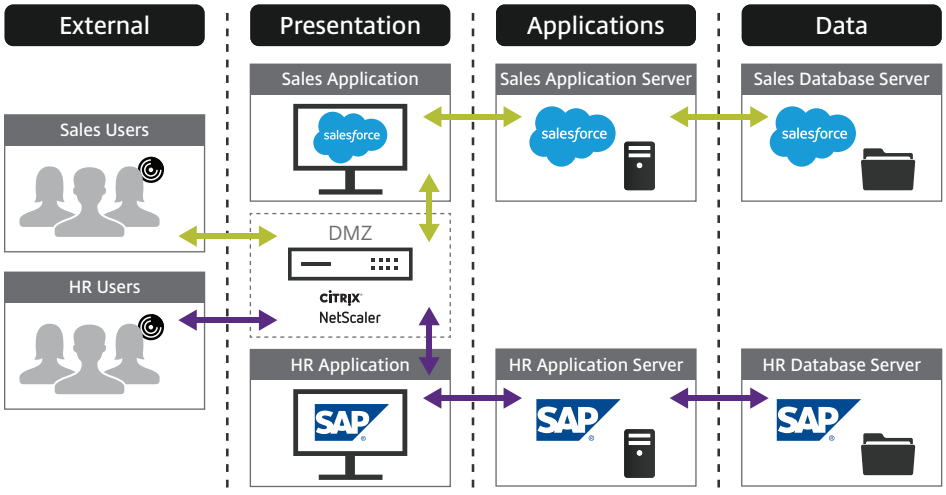
Additional steps may be needed, depending on which Microsoft SQL Server high availability solution is selected for the Delivery Controller.

Other links

Apart from the XenApp and XenDesktop communications paths shown above, you may need to protect other paths with TLS. As shown in the "Getting Started with Citrix XenApp and XenDesktop Security" white paper, there are communications paths between applications and databases to consider.

Link from Delivery Controller to hypervisor

This depends on the hypervisor. Refer to the vendor documentation for TLS configuration.



Other possible communications paths include from a web browser to a web server (as a published application in XenApp), or from a management console to a database server.

Consult the server's product documentation to determine its support for TLS. Then apply the general TLS principles described above to select an appropriate TLS configuration for the server.

Upgrading from earlier versions of XenApp

TLS 1.2 and TLS 1.1 are also supported in XenApp 6.5 using the Secure Gateway and SSL Relay components. Specific updates are required; refer to the Citrix Support Knowledge Center for details. For customers already planning to upgrade to XenApp 7.x and XenDesktop 7.x, this offers a smooth migration path.

Planning ahead: The future of end-to-end encryption

Information often needs to stay protected for many years. Cryptography is embedded in many places in large systems, and these places cannot all be upgraded at the same time. So cryptography needs long-term planning—10 years or more ahead.

Innovation in cryptography

Innovation in cryptography is unlike many other kinds of IT innovation. First, the mathematicians come up with a bright new concept. Then, the researchers try to turn it into something useful—such as a better way to do digital signatures. Next, the researchers take apart each others' work, refine their own, and the best proposals float to the top. Once there is market demand (usually because existing solutions are starting to get weak), there is then a public competition, with several rounds. The engineers build prototypes (software and hardware), and the cryptographers try to break everything. Eventually winning algorithms are chosen, with a detailed comparison of all the entries.

Now, the standards committees have to rework all the networking protocols to use a winning algorithm, and write this up in deathless prose. The regulators have to approve it; the legislators have to decide how it can be exported. Test labs need to update their test suites. The vendors have to turn the prototypes into production-quality software and hardware, and get them validated by the independent test labs. The vendors ship their products. Finally, customers update their security policies, and buy the products.

This whole process takes many years, and many people. Much of it happens in public. It may seem cumbersome—and some question it—but it delivers results.

Post-quantum cryptography

One important field of innovation is post-quantum cryptography.

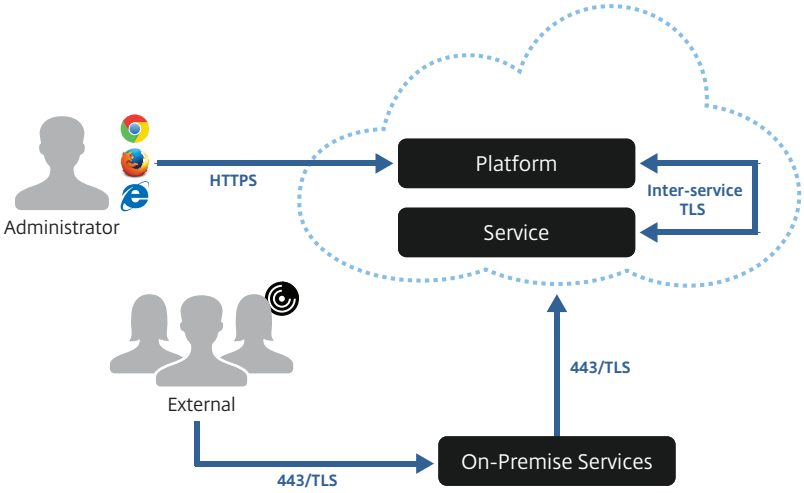
If a practical quantum computer can be built—and there are predictions that this could happen by 2030—it would defeat all the current digital certificate technologies: whether based on RSA, DH, ECDH, or ECDSA. If nothing were done, this would return the world to the mid-1970s, before the invention of public key cryptography.

So, a replacement is needed: this is post-quantum cryptography. There has been research for the last ten years for possible replacements, using different branches of mathematics. One or more will be selected within the next few years, using the kind of process just described.

The replacement will be integrated into a future version of TLS; it is too early to say which version. One thing is clear—the keys and digital certificates will be much larger than existing ones, so this will not be a simple drop-in replacement. The transition will be as tricky as that from RSA to ECDSA, for those who have already made that step.

The cloud today: Considerations for end-to-end encryption

Just as with the XenApp and XenDesktop deployments described earlier, end-to-end encryption with the cloud involves multiple links, which can be protected with TLS. The following diagram shows a top-level view.



Now that cloud computing is a familiar technology, and the business issues are understood, organizations are planning to migrate major systems to the cloud in their entirety. The vision: all such stuff into the cloud and leave not a rack behind.

Some cloud providers offer an HSM service to their tenants. This has the advantage that customers can control their own keys, and do not need to deploy their own hardware. The service may support key replication for availability, and provide an audit trail. The underlying hardware may have security certifications.

Consult your cloud provider for details of their HSM service. Check that it supports the applications you require (including TLS and DTLS), and the types of certificates you need (for example ECDSA certificates).

Intranet of things

Your network already contains many devices that need secure remote management; printers, access points, and others. Check whether they support TLS today, or an equivalent secure management protocol.

Internet of Things: End-to-end encryption

The Internet of Things (IoT) covers a very broad range: everything from complex devices with several ordinary computers embedded (such as a high-end printer), to an individual sensor using a custom silicon design for minimal battery usage.

End-to-end encryption is perfectly achievable for anything that has the capability of an ordinary computer, or even a low-end phone. Less powerful devices than that can still benefit from encryption as far as an IoT gateway. In the future, innovation in specialized lightweight cryptography (LWC) may extend end-to-end encryption as far as sensors.

Read the Citrix blogs to explore how Citrix Octoblu technology supports IoT security and encryption.

Summary: Effective use of end-to-end encryption

To make effective use of end-to-end encryption:

- Capture all your business use cases for end-to-end encryption, as determined by user needs, data sensitivity, regulation, compliance, contractual, or other business reasons
- Use hop-by-hop encryption where traffic inspection is required to protect against data exfiltration, malware infiltration, or other security reasons
- Select TLS configuration to match the actual business need. Do not enable older versions of TLS or SSL, or weak ciphersuites, "just in case" for compatibility
- Plan to review TLS configuration at least yearly, to remove deprecated options in order to stay in regulatory compliance
- Consider using NetScaler as a single point of enforcement for TLS policy—not just at the Internet boundary, but also at the boundaries of access layers in internal networks. Select the appropriate NetScaler product to match the deployment—for example NetScaler VPX at an internal boundary in a virtualized environment

With support for TLS 1.2, Citrix XenApp and XenDesktop provide end-to-end encryption for access to apps and data, whether deployed in the cloud or on-premise. Customers can select appropriate TLS ciphersuites to match their regulatory needs. This approach means that customers do not need to modify and retest the apps themselves to add TLS 1.2, but can continue to use existing apps. When used with NetScaler Gateway, TLS 1.2 support extends to remote users as well.

References

Getting Started with Citrix XenApp and XenDesktop Security

Exporting

Securing XenApp and XenDesktop Environments

Configuring the Secure Gateway or Secure Gateway Proxy

How Do I Configure Framehawk Support On NetScaler Gateway?

UDP Audio Through A NetScaler Gateway

Securing the Published Browser

Managing Certificates

XenMobile Security

ShareFile Enterprise: security white paper

Web Services Security Kerberos Token Profile 1.1

Citrix Octoblu is Securing the Internet of Things

Contributors

Steven Krueger, Lead Systems Engineer

Chris Mayers, Chief Security Architect

Kurt Roemer, Chief Security Strategist

Martin Zugec, Senior Technical Marketing Manager

Corporate Headquarters
Fort Lauderdale, FL, USA

Silicon Valley Headquarters
Santa Clara, CA, USA

EMEA Headquarters
Schaffhausen, Switzerland

India Development Center
Bangalore, India

Online Division Headquarters
Santa Barbara, CA, USA

Pacific Headquarters
Hong Kong, China

Latin America Headquarters
Coral Gables, FL, USA

UK Development Center
Chalfont, United Kingdom

CITRIX®

About Citrix
Citrix (NASDAQ:CTXS) is leading the transition to software-defining the workplace, uniting virtualization, mobility management, networking and SaaS solutions to enable new ways for businesses and people to work better. Citrix solutions power business mobility through secure, mobile workspaces that provide people with instant access to apps, desktops, data and communications on any device, over any network and cloud. With annual revenue in 2015 of $3.28 billion, Citrix solutions are in use at more than 400,000 organizations and by over 100 million users globally. Learn more at www.citrix.com.