Chapter 4

Understanding XenApp Security

Solutions in this chapter:

- **Defining the XenApp Security Model**
- **Defining Types of Deployments**
- **Understanding Authentication Methods**
- **Encrypting XenApp**

- ☑ **Summary**
- ☑ **Solutions Fast Track**
- ☑ **Frequently Asked Questions**

Introduction

Data security incidents and breaches into identity theft are now daily headlines. Until recently publicity regarding these breaches was more limited. In 2003, California was one of the first states to pass a law requiring companies to notify affected consumers regarding security breaches. Since then, many other states have passed similar legislation. These public notice requirements have heightened consumer and public awareness regarding data security breaches.

As a result of these many incidents, state and federal government agencies have created numerous policies that affect information security and the protection of data. Some of the most visible are the Sarbanes–Oxley Act of 2002, referred to as SOX, and the Health Insurance Portability and Accountability Act, referred to as HIPAA. Even though these policies identify what needs to be protected, they do not necessarily tell how to implement the protection. In some cases, administrators have found contradictory guidelines or guidelines that have caused a change in how information technology business is conducted. Additionally, some legislation only affects publicly held companies.

In many cases it is up to individual organizations to provide specifics for defining data security implementation. In this chapter, we hope to provide you with the information to take a manageable hold of your information security as it relates to XenApp.

The needs of your organization will dictate the optimal version of XenApp to use; for example, the requirements of your organization may dictate the need for session auditing, which only the Platinum Edition provides.

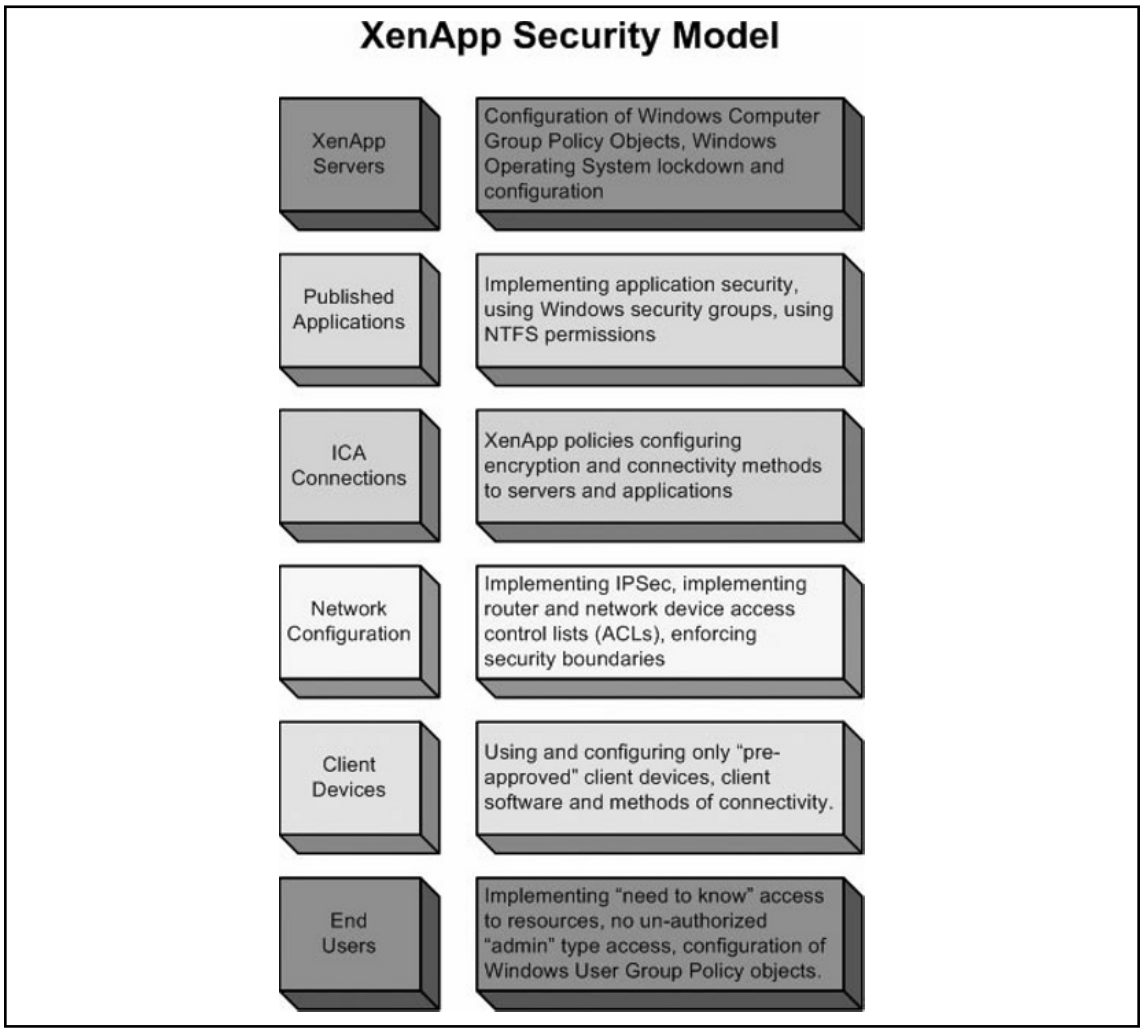
XenApp is offered in three versions:

- **Advanced Edition** This is the base version of XenApp that provides server-side application virtualization and application-based load management.
- **Enterprise Edition** In addition to the features of the Advanced Edition, the Enterprise Edition provides client-side application virtualization, application performance optimization, application compatibility optimization, and system monitoring and analysis.
- **Platinum Edition** This version includes all the features of the Enterprise Edition and provides additional features such as SmartAuditor session recording, application password policy control, universal SSL VPN with SmartAcess, application performance monitoring, and single sign-on application security.

Defining the XenApp Security Model

One of the first steps in securing your XenApp environment is to understand what we define as the *XenApp Security Model*. There has been some discussion on various forums and several documents have been prepared concerning how to secure Citrix XenApp Server. We will expand on that information and how it can be used to secure a XenApp environment. We have provided a culmination of these concepts into the XenApp Security Model. You can view the XenApp Security Model much like that of the *OSI Model*. The XenApp Security Model has six layers (see Figure 4.1).

Figure 4.1 The XenApp Security Model



As a XenApp administrator, your job is mainly ensuring that applications and resources are published for your users. Your responsibility is not only protecting your users and their data from malicious external threats, but also from internal malicious threats AND protecting them from inadvertent mistakes they can potentially make in an unsecured XenApp environment. Keep in mind that no matter how hard you go about locking down any environment (not just XenApp), there may be that one user that is able to defeat the measures that have been put into place. At one time or another, you have probably encountered the user that insists on proving that they know more than you do. It is a challenge for them to overcome the measures you have put into place to protect the network. There are also users that may end up bypassing security mechanisms simply by accident rather than by design. Implementing security based on the XenApp Security Model will assist you in protecting your network from these different threats and different users. You must carefully review each level at which your environment provides information to the user. To assist you in defining policies for your XenApp network, security should be implemented in the following manner:

- **XenApp Servers**
  - Implementing *computer* group policy objects
  - Locking down the base operating system
- **Published Applications**
  - Implementing application security
  - Using Windows security groups
  - Implementing NTFS permissions on application executables
- **ICA Connection**
  - Security on the ICA connection
  - Encryption
- **Network Configuration**
  - Using Internet Protocol security (IPSec)
  - Router ACLs
  - Enforcing security boundaries
- **Client Devices**
  - Using only approved client devices to connect
  - Using only approved client software to connect
- **End Users**
  - Establishing a "need to know" access for users
  - Not allowing any unauthorized "admin" type access
  - Implementing *user* group policy objects

You need to consider all aspects of your environment. Does a user really need the capability to run CSCRIPT.exe or CMD.exe? Does a user need the capability to surf the Internet? For a multihomed server, does the ICA protocol need to be listening on every interface? Is the firewall team going to think kindly of you when you want to open up Transmission Control Protocol (TCP) port 1494? Do you want to have your payroll application published on a server that also has applications available to all employees in your company? When applying security in the XenApp environment and by using the XenApp Security Model presented here, you will be better prepared to address situations like these.

NOTE

The XenApp Model presented here may not cover all aspects for your particular network. The model is intended as a starting point to help assist you in addressing the security concerns of your environment. You may have to add or remove elements to best suit the needs of your organization.

Defining Farm Security and Farm Boundaries

XenApp networks range in size, both in number of users and geographical area. Some XenApp farms may only have one server with 10 to 20 users. Other farms may have several hundreds of servers globally dispersed with thousands of users. Regardless of what type of environment you have, you should clearly define your environment to assist you with applying the XenApp Security Model to your network. Many administrators already have this task completed in the form of a *network diagram*. You may think that this is an unnecessary task in providing security to your environment, but the saying "a picture is worth a thousand words" holds true in the world of information technology and information security. By having a quick glance view of your network boundaries and assets, you can quickly ascertain your "weak" points of security. We have seen that a network boundary diagram does not necessarily have to show every single device in your network, but it should show a logical definition of your network.

Our example in Figure 4.2 shows a single-forest, single-domain, two-site corporate network. The corporation has grouped its externally available assets into a demilitarized zone (DMZ). You will notice that there are firewalls between each segment. The diagram does not go into details as to the number of servers at each location or the type or even the layers of protection provided by the firewalls. For example, the firewall could be a combination of an actual firewall device and router ACLs defining access, but our diagram simply shows it as a boundary. So, what do we define as a boundary? In the simplest of definitions, a *boundary* can be defined any time data traverses a network device or leaves one logical network and enters another. Many corporations have multiple internal network devices that can bridge virtual LANs (VLANs) or provide redundancy, so would these be considered network boundaries? It is up to you, as the administrator, to determine how to best group these devices to come up with a workable network boundary diagram that you can use to assist you with identifying potential problems.

Once you have identified a workable network boundary diagram, simply add your XenApp components into the mix. In our example network, the corporation wants to make MS Office applications available to its users both internally and externally. The corporation uses smart card technology for internal authentication and wants to also use that same authentication mechanism for external users. The corporation is also concerned about the data because it contains sensitive information, so they want end-to-end encryption. The resulting XenApp farm boundary diagram is provided in Figure 4.3. The Web Interface server and a Secure Gateway server are placed in the corporate DMZ and the XenApp servers are placed at each site. The corporation decides that internally, the data is protected enough via Basic encryption provided by XenApp and that the encryption of the links between the sites and the corporate DMZ mitigates the need to further encrypt the data between the DMZ and the corporate sites. However, they do want external access encrypted to the maximum extent possible. How to go about configuring the scenario is explained in detail later in this chapter, but we have added it to our final XenApp network boundary diagram in Figure 4.3. From this diagram, the network administrator should be able to quickly ascertain where the potential for security risks lies.

WARNING

Having a network and farm boundary diagram can be extremely useful. But be careful as to the specific information that you place on the document. For example, do you want to include IP addresses for network devices and servers on a displayed

diagram if your diagram is displayed where outside personnel, such as temporary contractors or consultants, could potentially see the information? The bottom line here is to assume that if you are advertising the information where someone can see it, then they can also remember that information for later use.

Figure 4.2 Defining Network Boundaries

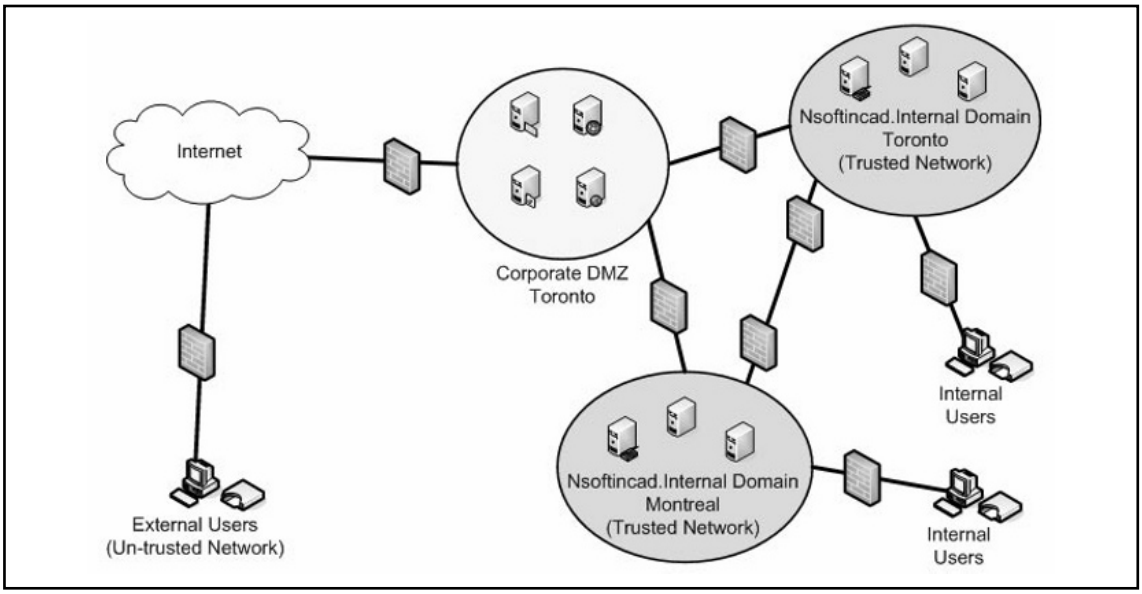
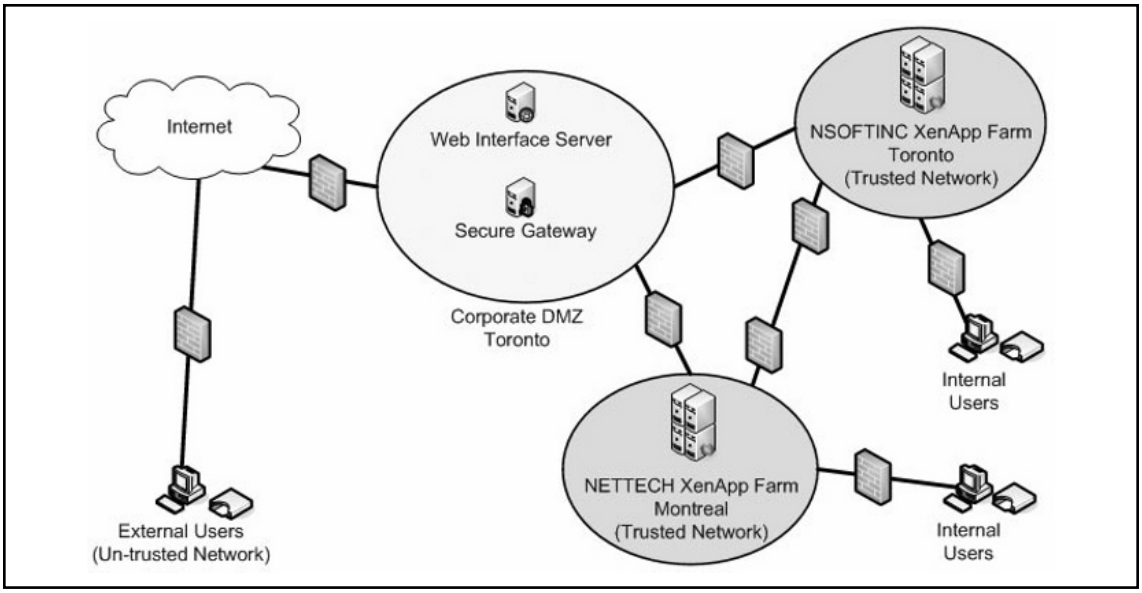


Figure 4.3 Defining XenApp Farm Boundaries



Defining XenApp Server Security

To protect vital information from unauthorized intruders, it is vital that you secure your network and computer assets. As computer and network systems have become more common, the need for security has grown exponentially. As an administrator, you must be careful to ensure that you take into account every option that can assist in securing the computing environment. Although Citrix XenApp provides several methods to ensure security of vital information, other products and solutions are used to protect data throughout a computing environment.

A critical component of XenApp security is the security of the underlying operating system (OS) platforms on which the XenApp software runs. If it is not possible to secure the OS, then XenApp itself cannot be secure. Even a securely configured operating system is vulnerable to the flaws of the programs and applications that run on it.

Introducing Microsoft Security Tools

The first level of the XenApp Security Model deals with the server itself. First and foremost, you need to have your Windows server properly configured and locked down. There are many ways that you can secure the base operating system. Microsoft has many freely available tools that can assist you with the security configuration of your servers and help you to maintain an effective security posture, such as:

- **Security Configuration and Analysis Tool** This is a Microsoft Management Console (MMC) snap-in that allows you to use default or custom configured templates so that you can analyze and configure security settings on a Windows 2003-based computer.
- **Microsoft Baseline Security Analyzer (MBSA)** This tool, shown in Figure 4.4, scans for missing security updates and common security settings that are configured incorrectly. Typically, this tool is used in conjunction with Microsoft Update or Windows Server Update Services.
- **Extended Security Update Inventory Tool** This tool is used to detect security bulletins not covered by the MBSA and future bulletins that are exceptions to the MBSA.
- **System Center Configuration Manager** This tool provides operating system and application deployment and configuration management. This is the latest version of Systems Management Server (SMS) 2003.
- **Microsoft Security Assessment Tool (MSAT)** This tool, shown in Figure 4.5, is designed to help you assess weaknesses in your information technology (IT) security environment. The tool provides detailed reporting and specific guidance to minimize risks it has identified.
- **Microsoft Update** (www.update.microsoft.com) This Microsoft Web site combines the features of Windows Update and Office Update into a single location that enables you to choose automatic or manual delivery and installation of high-priority updates.
- **Windows Server Update Services (WSUS)** This tool provides an automated way for keeping your Windows environment current with the latest updates and patches.
- **Microsoft Office Update** (www.officeupdate.com) This Microsoft Web site scans and updates Microsoft Office products.

- **IIS Lockdown Tool** This tool provides security configuration for Internet Information Servers (IIS) and can be used in conjunction with *URLScan* to provide multiple layers of protection against attackers.
- **UrlScan Tool** This tool helps prevent potentially harmful HTTP requests from reaching IIS Web servers.
- **EventCombMT** This multithreaded tool will parse event logs from many servers at the same time to assist you with finding specific event entries.
- **PortQry** This tool is a Transmission Control Protocol/Internet Protocol (TCP/IP) connectivity testing utility that can aid you in determining active TCP ports in use on a system.
- **Malicious Software Removal Tool** This tool checks a system for infections by specific, prevalent malicious software, to include *Blaster*, *Sasser*, and *Mydoom*. The tool can also assist in the removal of any discovered infections. Microsoft releases an updated version of this tool every month.
- **Port Reporter** This tool is a service that logs TCP and User Datagram Protocol (UDP) port activity.

Figure 4.4 Using the Microsoft Security Baseline Analyzer Tool (MBSA)

Figure 4.5 Using the Microsoft Security Assessment Tool (MSAT)

TIP

The servers used in our labs for this book were first locked down using the Windows Server 2003 Security guide and the tools listed above available from the Microsoft Web site, www.microsoft.com/security. The National Security Agency (NSA) also provides several documents that are publicly available from their Web site, www.nsa.gov/snac, to assist in the securing of other assets. As in any environment, you should first implement the recommended settings in a test environment before configuring a live production network.

Understanding Alternate Data Streams

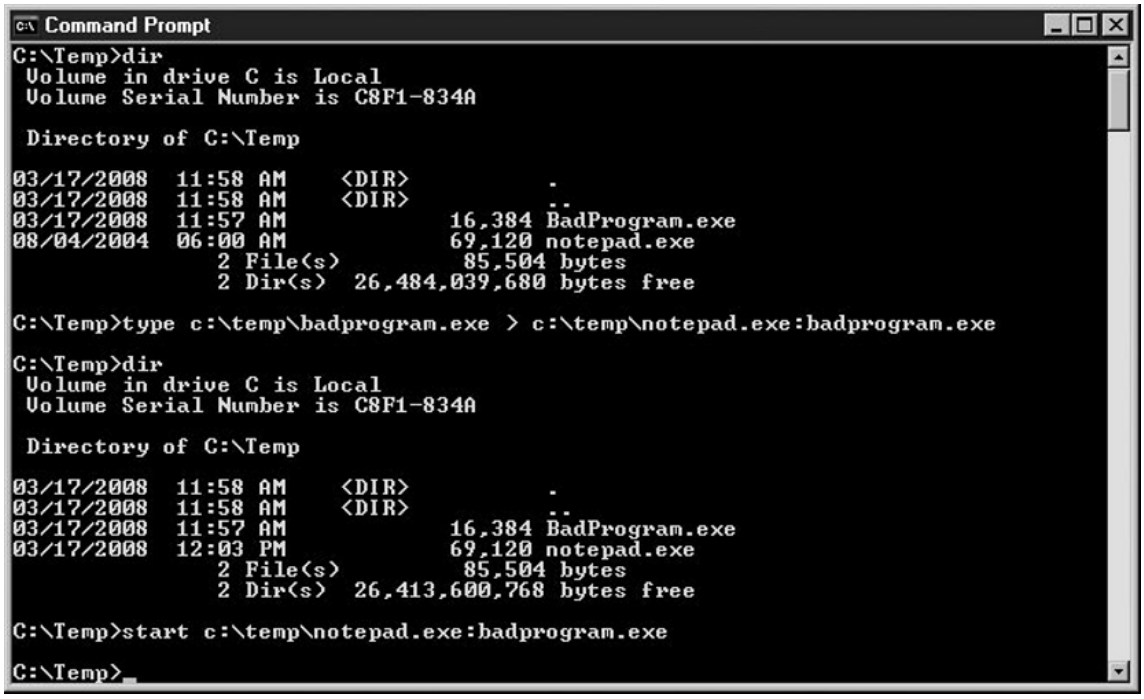
Alternate Data Streams (ADS) is a virtually unknown compatibility feature of New Technology File System (NTFS) that can provide attackers with a method of hiding hacker tools, keyloggers, and so on, on a breached system and then will allow them execution without being detected. You need to be aware that an attacker does not play by any rules. Nothing is off limits when attempting to breach a system. In so doing, attackers have become very adept at hiding their tracks. Why does ADS exist? ADS capabilities were originally designed to allow for compatibility with the Macintosh Hierarchical

File System, HFS, where file information is sometimes inserted, or forked into separate resources. ADS is used for legitimate purposes by a variety of programs including the Windows operating system to store file attribute information and for temporary storage. Directories can also support ADS.

Typically the task of copying a root kit or other hacker tools can be tricky with the products that are installed in most environments, but an attacker that knows how to exploit ADS can be successful if proper security measures are not exercised. You should never underestimate the determination of someone that truly wants to breach your system.

A popular method that attackers use for covering their tracks on Windows-based systems is the use of ADS. The use of ADS provides the capability to store one file in another without outwardly changing the appearance, functionality, or size of the original file. The only modification is the file date, which can be changed by freely available utilities. In Figure 4.6 we have two programs listed, NOTEPAD.exe and BADPROGRAM.exe (a sample hacker tool). The figure illustrates the original states of the files. Then we insert the file BADPROGRAM.exe into NOTEPAD.exe by using the following command: **type c:\temp\badprogram.exe > c:\temp\notepad.exe:badprogram.exe**. Following along in the figure you will notice that the only thing that has changed about the original file NOTEPAD.exe is the file date. At first glance there is really no way to determine if a file is utilizing the ADS feature. Inspecting the file through a command prompt or Windows explorer does not give you any hint that the file has been modified other than the time stamp.

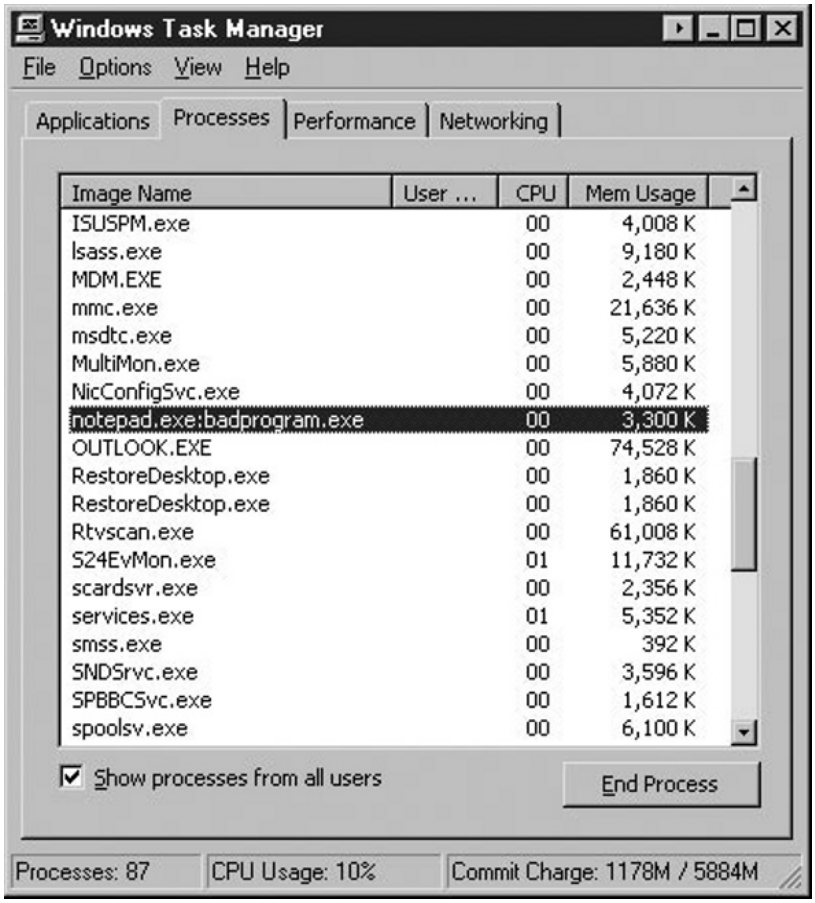
Figure 4.6 Using Alternate Data Streams



The next line in the figure shows how the inserted program can be executed by entering: **start c:\temp\notepad.exe:badprogram.exe**. Running Task Manager now reveals that the file is using ADS as shown in Figure 4.7. Older versions of Windows did not show this and the issue of ADS was

even more of a concern because damaging processes could then be executed without fear of detection. Only the most robust of intrusion detection systems will be able to identify and warn of files or processes initiated through an ADS. Moving an ADS to another system that supports ADS will keep the ADS file intact; however, if the file is moved to a system that does not support ADS, then the ADS is automatically destroyed.

Figure 4.7 File Using an Alternate Data Stream



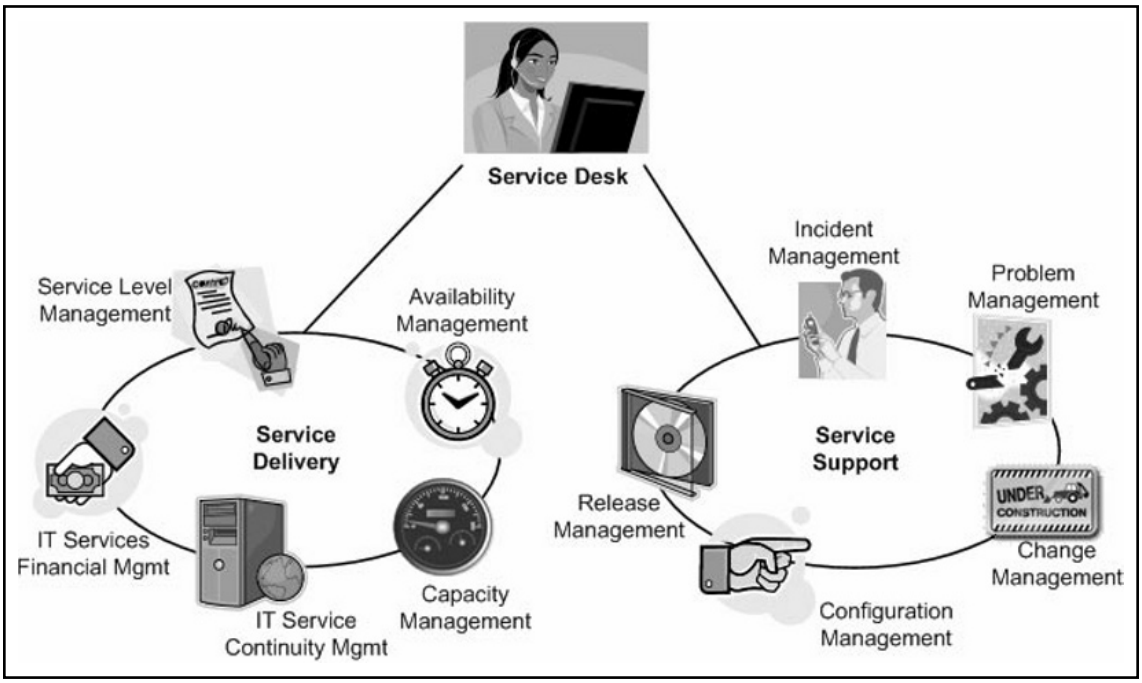
Understanding Security Configuration and Remediation

Security configuration and maintenance at the server level can be a very demanding and time consuming task, even for a small environment. Many organizations have adopted the Information Technology Infrastructure Library (ITIL) methodology for IT management. ITIL is a collection of guidelines and techniques for managing IT infrastructure, development, and operations, shown in Figure 4.8. ITIL covers areas such as configuration management, change management, and security. Implementation of this initiative can prove to be invaluable for your organization. To help implement some of the recommendations presented in ITIL there are many third-party tools that can assist with security configuration

compliance scanning, security compliance remediation, configuration management, etc. We have listed just a few software packages and their descriptions that can support the parts of the ITIL methodology.

- *BladeLogic Operations Manager* performs, among other things, patch management, compliance measurement, enforcement, and reporting.
- *HP Data Center Automation Center* is a suite of products that can perform patching, configuration management, script execution, compliance assurance, incident resolution, change orchestration, and many other tasks in a standardized and documented manner to enforce ITIL and compliance.
- *BMC Performance Manager for Servers* provides server monitoring, process monitoring, log file monitoring, and Windows event log monitoring. (BMC has a product specifically for Citrix called *BMC Performance Manager for Citrix Presentation Server,* but this product deals primarily with Citrix performance monitoring and optimization.)
- *IBM Tivoli* products such as *Compliance Insight Manager* that provides effective, automated user activity monitoring through high-level dashboard and compliance reporting, *Risk Manager* that manages security incidents and vulnerabilities, *Security Compliance Manager* that identifies security vulnerabilities and security policy violations, *Security Information and Event Manager* that provides a centralized security and compliance management solution, and *Security Operations Manager* that is designed to improve security operations and information risk management.

Figure 4.8 Understanding the ITIL Methodology

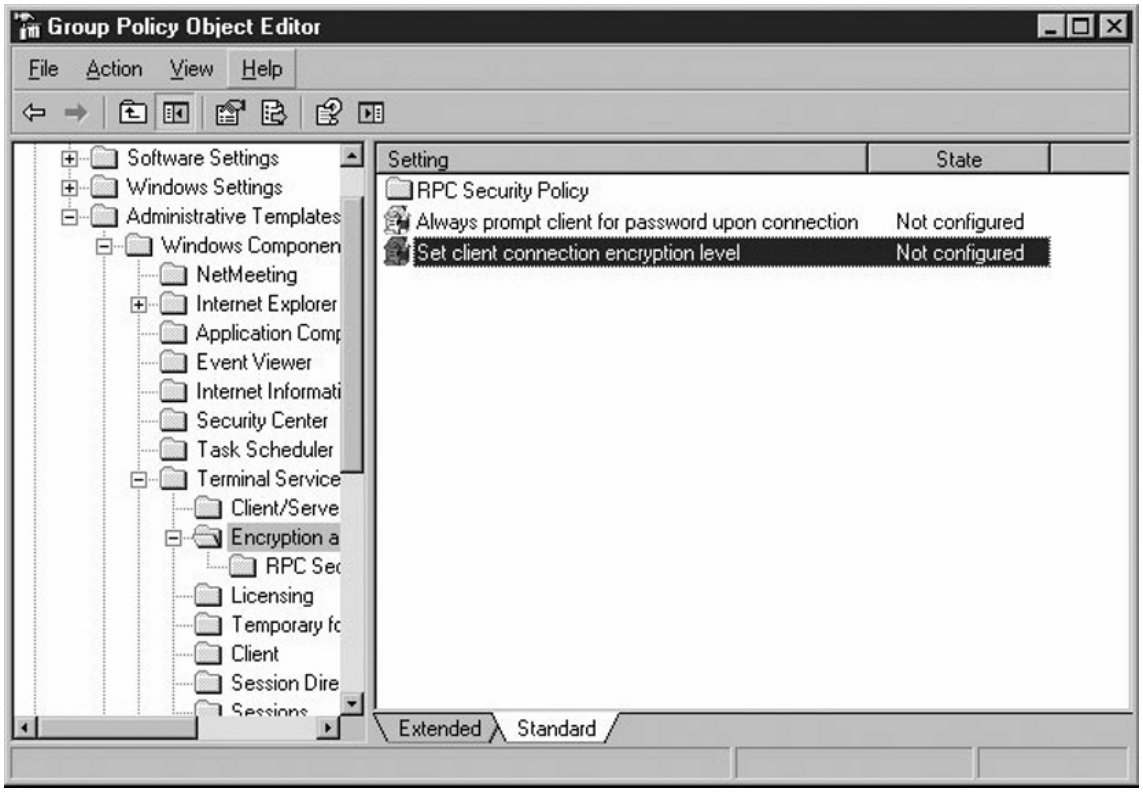


Using Windows Group Policy Objects

Your XenApp environment can be further configured by the use of *Group Policy Objects (GPOs)*, *Domain Security Policies*, and *NTFS permissions*. There are many different levels of the group policy object, but the basic details are shown in Figure 4.9. Group policies can be set by domain or be set locally on a server, or in some cases can be set in both places. Because of the flexibility offered with creating Windows GPOs, you could segregate your XenApp assets into their own Organizational Unit (OU) and apply specific GPOs that configure different aspects of your XenApp environment. You could have one GPO that deals specifically with server configuration settings for XenApp servers. Another GPO could be created to deal only with user level configuration settings on the server.

Your environment may already make use of default domain policies that apply to all servers. For example, settings such Server Message Block (SMB) timeouts, screensavers, and warning banners may apply to every server regardless of function, so be careful not to introduce a setting in a GPO that will adversely affect an unintended system. Terminal services settings in GPOs have been shown to affect both the RDP and ICA protocols, so having a separate GPO for remote administration and one for XenApp servers would be prudent. Can you imagine the result of having a single GPO for all of your terminal servers only to then have another administrator set the terminal services maximum connection time setting to three hours? This could prove disastrous if you are providing vital applications for users that stay logged on most of the day.

Figure 4.9 Using Window Group Policy Objects



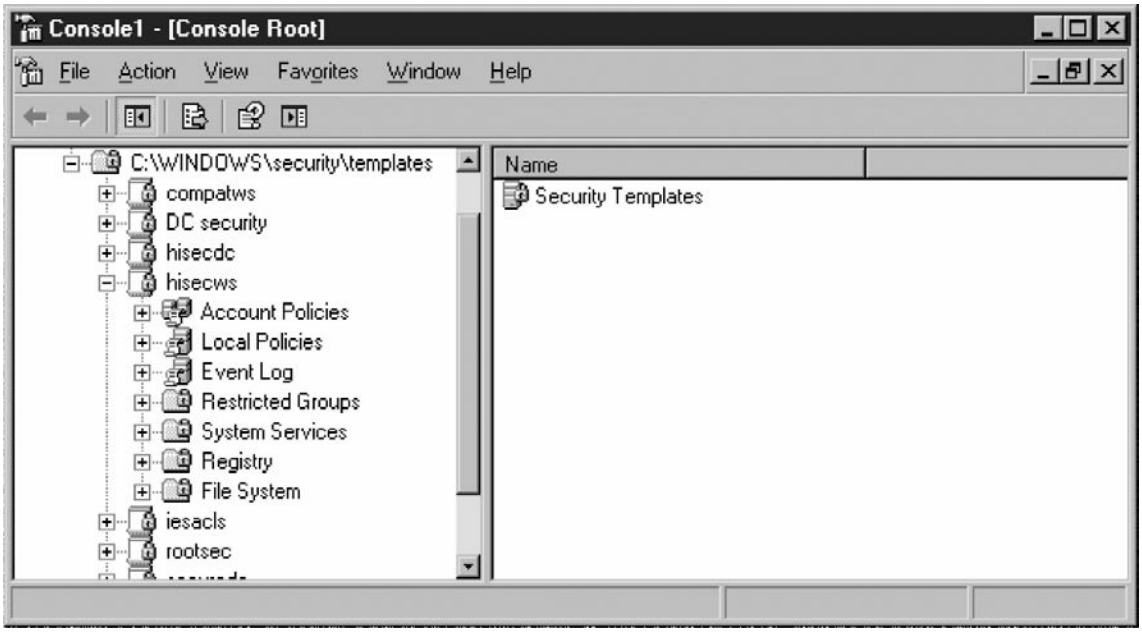
Using Windows Security Templates to Baseline Your System

As we have shown, Microsoft has many different solutions to assist you in securing your base operating system. But what happens if you have a customized set of security policies you would like to set that is beyond what is available through a GPO? The answer is Windows security templates. Security templates are not a new concept; they have been in existence since Windows NT. A *security template* allows you to configure security settings for different types of computers that you predetermine. You can view and configure existing Microsoft security templates through the MMC by selecting the **Security Templates** snap-in, shown in Figure 4.10.

WARNING

You can create a custom security template to be used in configuring your system. Keep in mind that designing a custom security template is just like making a change to your registry. If you make a mistake in the creation of a custom template, then you can adversely affect the performance and availability of your system once the custom template is deployed. Care and diligence should be exercised when creating a custom security template.

Figure 4.10 Utilizing Security Templates



Upon configuration of a security template you can analyze your system utilizing the **Security Configuration and Analysis** MMC snap-in and your newly configured template. Using security templates in conjunction with the **Security Configuration Wizard** can provide you with an excellent way to configure your system and then to periodically check it for changes from its original security configuration.

Defining an Antivirus Solution

Your XenApp environment should also utilize *antivirus* software. By definition, a *virus* is a piece of computer code that produces unwanted results; it has the unique ability to replicate itself. A virus can perform an amazing array of damage, ranging from annoying messages and extensive resource utilization to destroying files and systems and causing massive outages. In addition, viruslike programs known as worms have become more prevalent due to their potential impact. The ability to protect computers against these types of attacks has become more a necessity than a luxury.

When considering antivirus software in your Citrix XenApp environment, you must take into account several factors. First, you must evaluate the various products along with feature sets to provide a solution to meet your organization's needs. It's very important to ensure that the software you select is supported in a Terminal Server and Citrix XenApp environment.

In addition to using antivirus software on your Citrix XenApp servers, you can use various products throughout the network to protect other resources available to a Citrix client, such as file servers, electronic mail, and Internet Web filtering. Limiting your user's capability to surf the Web or use e-mail can also reduce the risk of virus infection. When you use antivirus software on your Citrix XenApp server, you must carefully configure the application to minimize the impact to end users. Most antivirus solutions provide real-time scanning of file access, but you must carefully consider its impact on server performance. Carefully test how this software impacts the overall client experience to ensure that it's not causing more damage than good. In addition, active scanning can be performed to search the entire system for any virus. Although this is can be an effective tool, you should test the effect of implementing its use during peak production hours because it could cause severe performance degradation.

Last, antivirus software uses signatures to identify virus patterns while scanning. To ensure you are monitoring for the latest virus infections, you must periodically update the signatures from the manufacturer. Most software solutions available offer scheduled automatic updates. In addition, you can manually update the signature files if needed. It is recommended that you determine an acceptable interval for updating your antivirus signatures. You should check for new signatures at least once a week and install updates during nonpeak hours to minimize any user impact. In addition, if you become aware of any new virus infections, immediately check the vendor's web site for signature updates and information about the infection.

One of the most common threats today, virus attacks produce an astounding impact on organizations. With estimated damages being reported in the billions of dollars by various news sources, virus protection is a critical component to ensure that your networking environment is secure. Antivirus programs created by third–party software developers have become a huge part of any organization's security program.

Understanding Intrusion Detection

Another security measure you must consider is monitoring for intrusion. As hackers become more prevalent and savvy, you need additional tools to help protect your network environment. Intrusion detection is a strategy that any organization must consider.

Intrusion detection can be defined as the ability to monitor and react to computer misuse. Many hardware and software products on the market today provide various levels of intrusion detection. Some solutions use signatures to monitor for known attacks. Some platforms provide network monitoring; others are host-based systems. Some solutions react to particular alerts by shutting down services; others use a more passive approach. You must carefully select an intrusion detection strategy to ensure that your network resources remain secure from unwanted trespassers. Similar to virus protection, various locations and methods are appropriate to for intrusion detection. The most common use is to install an intrusion detection solution to monitor the access points from the Internet or outside world into your private networks. There are two main types of solutions: network-based and host-based. *Network-based* intrusion detection monitors network traffic for particular signs of malicious behavior. For example, if a user is continually trying to access a port known to be used with worms or Trojan horses, that could trigger an alert. *Host-based intrusion* detection programs are software products that are installed on your servers to monitor for suspicious behavior. This solution watches for viruslike activity so it can be stopped before it infects anything. It is critical to determine where you should monitor for intrusion and provide the appropriate solution to achieve these goals.

WARNING

The improper configuration of a host-based intrusion detection system can produce unwanted and adverse results. The worst can cause your XenApp server to be unavailable to your users. Care and diligence should be exercised when configuring a host-based intrusion detection policy for your environment.

Are You Owned?

Determining a Baseline for Your System

Having a comprehensive *baseline policy* is essential in the management of any system. Microsoft provides some good tools, such as Microsoft Baseline Security Analyzer, that you can use to evaluate your server for inconsistencies. Why is having a baseline important? A baseline serves as the starting point for measuring future changes to a system. The baseline indicates a state at a certain point in time; the result of changes made to a system (such as patches and hotfixes) can help determine from that point forward if

Continued

the overall performance and security health of the system is improving, staying even, or getting worse. Maintaining an effective baseline for your environment should be an essential part of your organization's *change management process*. A baseline can also be used to determine any unauthorized changes to your system.

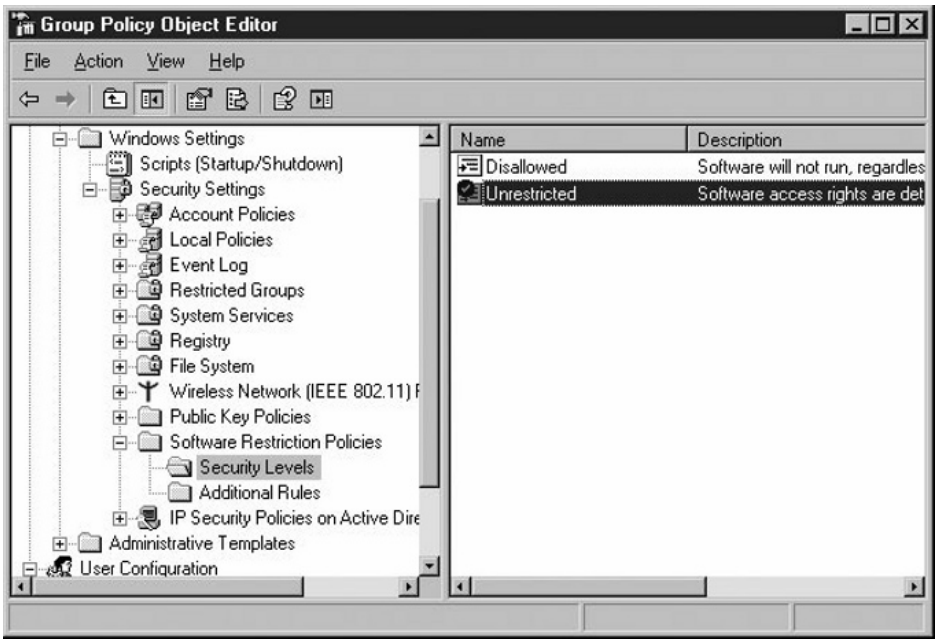
Understanding Published Application Security

When XenApp is installed on a server with the default settings, it can allow XenApp users to run virtually any application they choose on the server. For security purposes it is necessary to limit the applications that a XenApp user can run on the server. Windows provides administrators the ability to control user access to applications in a number of ways. Group Policy can specify which applications are visible to a user. In addition, it can prevent users from launching applications through the Windows Explorer shell. However, users can launch hidden applications either by using the Run command or by launching an embedded object. Group Policies can affect a user's computer as well as their XenApp sessions if not created properly. The recommended approach to limiting access to applications is to implement Application Security policies, NTFS security, and XenApp application policies.

Understanding Application Security Policies

In Windows 2000, the Microsoft Application Security tool (APPSEC.exe) could allow an administrator to control access to each application/executable on the server. If configured properly, it could limit a user to accessing only specific applications via a XenApp ICA session. With Windows Server 2003, you now have *software restriction policies*, as shown in Figure 4.11, that can accomplish the same thing as APPSEC.exe and are much more flexible to use.

Figure 4.11 Using Software Restriction Policies



WARNING

Some people believe that providing access only to a single application (through published applications with XenApp) provides some measure of security. This is not the case. Even if you have configured only one application, such as NOTEPAD.exe, the unsecured environment can be breached. Any reasonably intelligent user will quickly learn that **Ctrl-Alt-Esc** will launch Windows Security in their session, which could potentially be a back door into your system.

When using software restriction policies, you can identify and specify the software that is allowed to run so that you can protect your XenApp environment from unauthorized programs and files. The basic configuration has two policies: *Unrestricted*, which allows access to software based on the access rights of the user and *Disallowed*, which prevents the execution of software regardless of user access rights. For your XenApp environment, you must select one of these policies as the default and then fine-tune the policy for your specific environment. To do this, you create exceptions to your default security level with rules. You may create rules such as the following:

- Hash rules
- Certificate rules
- Path rules
- Internet zone rules

A policy is made up of the default security level and all of the rules applied to a group policy object. You may apply the policy to specific computers or users. Implementing application security via the use of software restriction policies provides a number of ways to identify what software can be executed in your environment.

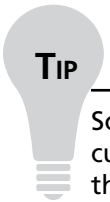
With software restriction policies, you can control which programs can run on your computer. You can permit users to only run certain file types. However, in large environments with several hundreds of servers and published resources, the administration of software restriction policies can prove to be arduous.

WARNING

Even though you can use software restriction policies to prevent the execution of specific files, Microsoft recommends that you do not use software restriction policies as a replacement for antivirus or antispyware software.

Explaining NTFS Permissions for Published Application Security

Another way to restrict applications that a user can access is to configure NTFS security on the application executables themselves. Using NTFS-level security pretty much ensures that the access restriction will not be circumvented. With NTFS security, no matter how the user accesses the server, the application will not run if the user does not have the proper NTFS rights to the application. The catch here, just like with using software restriction policies, is management of the security settings.



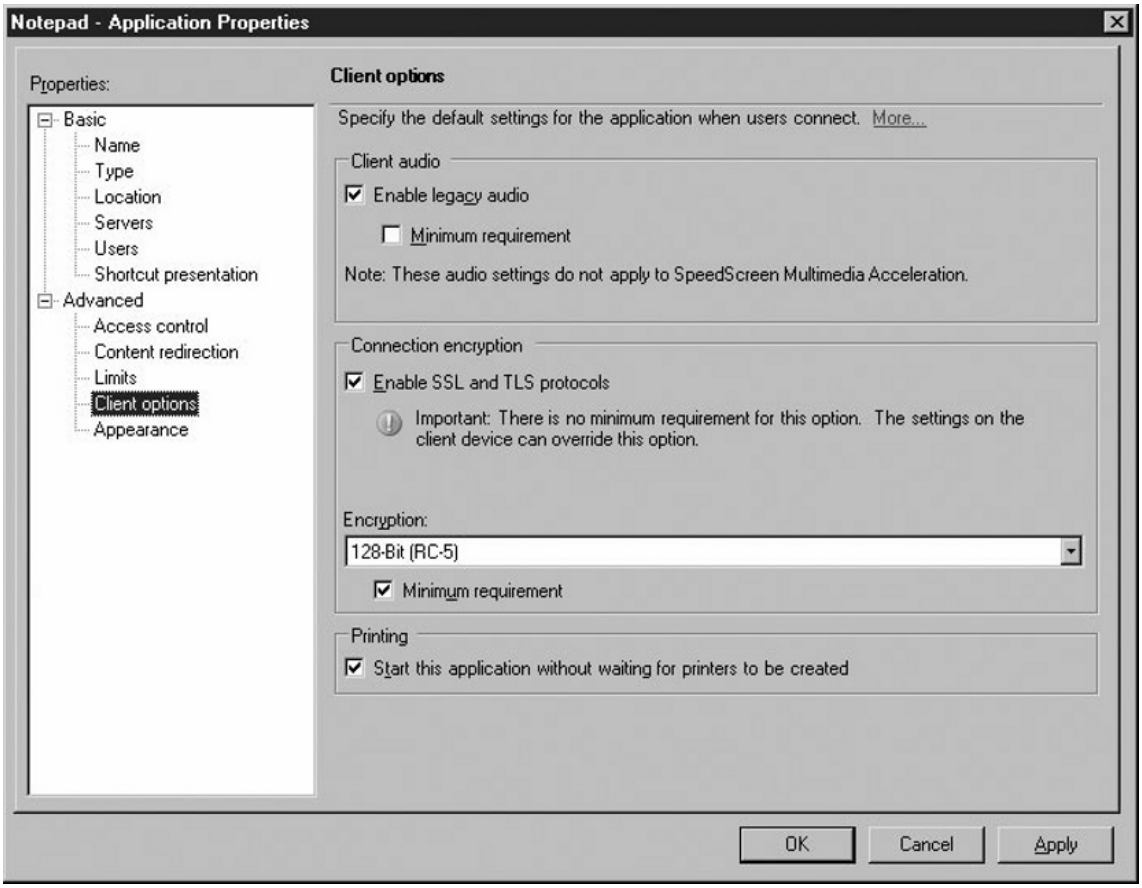
TIP

Some administrators find the maintenance of application security policies cumbersome and challenging. There are several third-party products that accomplish the same result, but have a much more robust user interface and are easy to use. One such product is Tricerat's Simplify Lockdown application that is part of Tricerat's Simplify Suite v4 server management tools for terminal server-based environments.

Defining XenApp Published Application Properties

You can define settings on each individual published application by selecting to modify the published application's properties, shown in Figure 4.12. You can select to have an application only be accessed via a particular encryption level, or to enable Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols. You can choose the methods in which the application can be accessed, such as utilizing a Citrix Access Gateway which can be configured with its own security policies. You can place limits on the number of connections and even select to temporarily disable an application from being available if necessary.

Figure 4.12 Using XenApp Published Application Properties



Notes from the Underground…

Who Wrecked My Server?

Having a system *audit policy* defined is a great place to start for tracking unusual events, because it can track account logons, account management changes, privilege use, etc. The Microsoft EventCombMT tool allows you to parse event logs from many servers at the same time to assist you with finding specific event entries, like repeated log-on failure events. Another tool is SmartAuditor, which is a new feature of Citrix XenApp Platinum Edition that uses policies to initiate recordings of user sessions. With SmartAuditor you can examine and monitor user activity.

Understanding ICA Connections

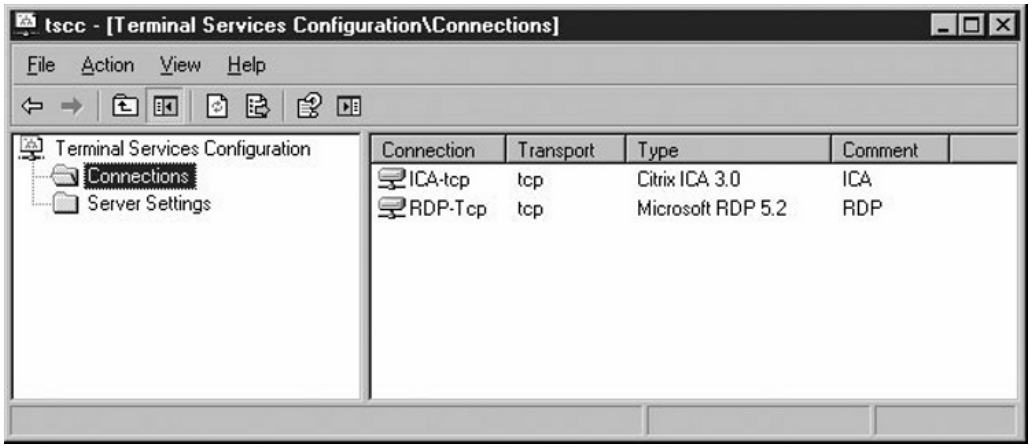
The ICA connection is the connection between the user and the XenApp server. Specifically, this is the *ICA protocol*. The ICA protocol can provide full client device mappings such as stereo, Component Object Model (COM) and printer mappings, and local drive remapping. What makes ICA so robust is its small size. The ICA protocol provides graphical and input information between the remote client and the XenApp server. Only screen updates and input are passed to lessen the bandwidth requirements to the client.

You can provide security on the ICA connection by:

- Encrypting the connection.
- Placing custom NTFS security on the protocol.
- Limiting the number of connections that use that protocol.
- Isolating the network card on which the protocol is bound on a multihomed server.

With Citrix Presentation Server 4.0 and earlier, you could configure the ICA protocol with the *Citrix Connection Configuration utility* (MFCFG.exe). On XenApp, the functionality of the Citrix Connection Configuration is added to the *Terminal Services Configuration utility* (TSCC.msc) as shown in Figure 4.13. Using this utility, you may lockdown specifics of the ICA protocol. Many features of the ICA connection are also configurable through the use of XenApp policies.

Figure 4.13 Using Terminal Services Configuration Utility



Understanding Network Configuration

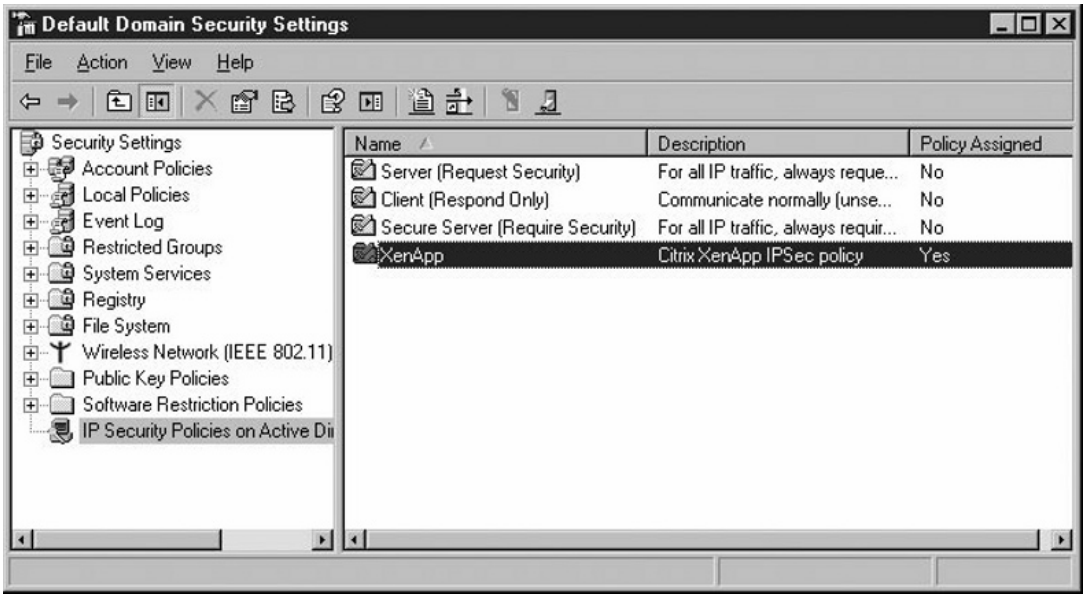
Earlier in this chapter we defined the need for defining network boundaries and using a network diagram to assist in this task. Using a network diagram will help you in determining what needs to be secured and how you should go about securing it. Security at the network level for your XenApp environment can be accomplished by the following means:

- Implementation of IP security
- Implementation of router and network device *access control lists (ACLs)*
- Segregation of XenApp components based on role
- Disabling unused and unneeded ports, protocols, and services on the XenApp server

Internet Protocol security, or IPSec, is a framework of open standards for ensuring private, secure communications over IP networks, through the use of cryptographic security services. The following are Microsoft best practices for implementing IPSec:

- Establish an IPSec deployment plan.
- Create and test IPSec policies for each deployment scenario.
- Do not use preshared keys.
- Do not use Diffie–Hellman Group 1 (low).
- Use the Triple Data Encryption Standard (3DES) algorithm for stronger encryption.
- Create and assign a persistent IPSec policy for failsafe security.
- For computers connected to the Internet, do not send the name of the *certificate authority* (CA) with certificate requests.
- For computers connected to the Internet, do not use Kerberos as an authentication method.
- For computers that are connected to the Internet, do not allow unsecured communication.
- Restrict the use of administrative credentials in your organization.
- When applying the same IPSec policy to computers running different versions of the Windows operating system, test the policy thoroughly.
- Use the Windows Server 2003 IP Security Policy Management console (shown in Figure 4.14) to manage the IPSec policies that use the new features that are available only in the Windows Server 2003 family implementation of IPSec.
- Use Terminal Services to remotely manage and monitor IPSec on computers running different versions of the Windows operating system.

Figure 4.14 Using the IP Security Policy Management Console

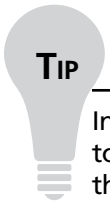


When connecting your computing resource to other networks such as the Internet, you must consider how you control access between your network and the others. In addition, some resources within your organization (such as human resources department computers that hold confidential personal information) need to be secured from internal intruders. A *firewall* is a common technique used to meet these requirements. A firewall is traditionally used to secure one set of network resource from another network. The most common implementation of firewalls today is organizations connecting their internal private networks to the Internet. A firewall allows administrators to restrict outside individuals' access to internal resources. Although many firewall products are on the market, a firewall is more a security strategy than a single product. The solution to fit your needs might be available in a single product, but many times multiple devices are required to completely secure a network. For example, many firewall implementations include items discussed earlier in this chapter, such as intrusion detection and IP security. As an administrator, you must select the options and product that best suit your environment's requirements. In order to utilize Citrix XenApp with firewalls, you must carefully consider who and what resources need to be available to external users. Whether it is you or another team that is responsible for your organization's firewall configuration, items discussed earlier in this chapter (network diagrams, PortQry, Port Reporter, etc.) will prove to be invaluable resources in the configuration.

WARNING

Incorrectly configuring IPSec policies can deny clients and other servers access to your system. You should have a thorough and complete understanding of your network environment before applying IPSec policies. Optimally, you should test IPSec policies in a test environment before applying on a production system.

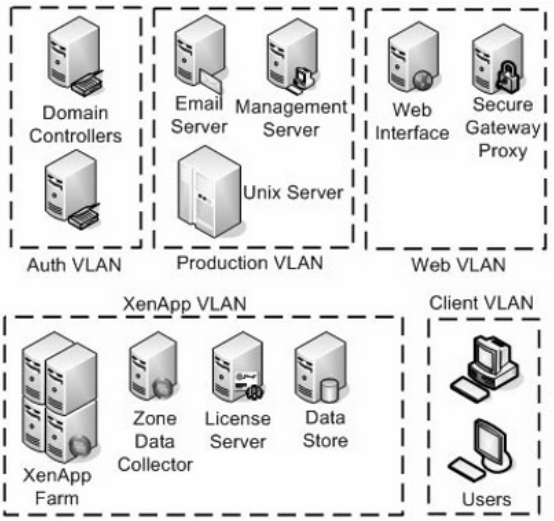
Segregation of your XenApp resources is essential to a good security posture. By segregating your environment in this manner you can isolate potential problems to specific areas of your network instead of having the entire network affected. In Figure 4.15 we have placed our XenApp assets into a separate virtual LAN (VLAN) called XenApp. There are also VLANs defined for clients, web, production, and authentication. You can further secure the connectivity between servers with the implementation of IP Security policies, especially useful for protecting your XenApp license server. You should place any computer that is directly connected to the Internet in a DMZ and these servers should be stand-alone servers that are not part of a domain. Another way of providing additional security is by segregating traffic within your XenApp network by using multiple network cards in your servers. For example, you could have one network card that is used exclusively for private communication between XenApp resources and domain authentication, another card could be used for public communication between your XenApp resources and resources that are directly connected to the Internet, another network card could be used for conducting backups of your assets, and yet another card could be used for remote administration of your XenApp server. Establishing a multihomed server in this fashion requires the configuration of additional resources and potentially the configuration of a separate VLAN for each network card. However, by implementing a multihomed server in this fashion could provide increased performance by not overloading a single network card with multiple tasks.



TIP

In some configurations that make use of a multihomed server, you will find it necessary to ensure that the XenApp server communicates on the proper network card. To do this, you can use the command line **Altaddr** utility to configure the IMA and ICA Browser services to return the alternate/external IP address to the XenApp environment.

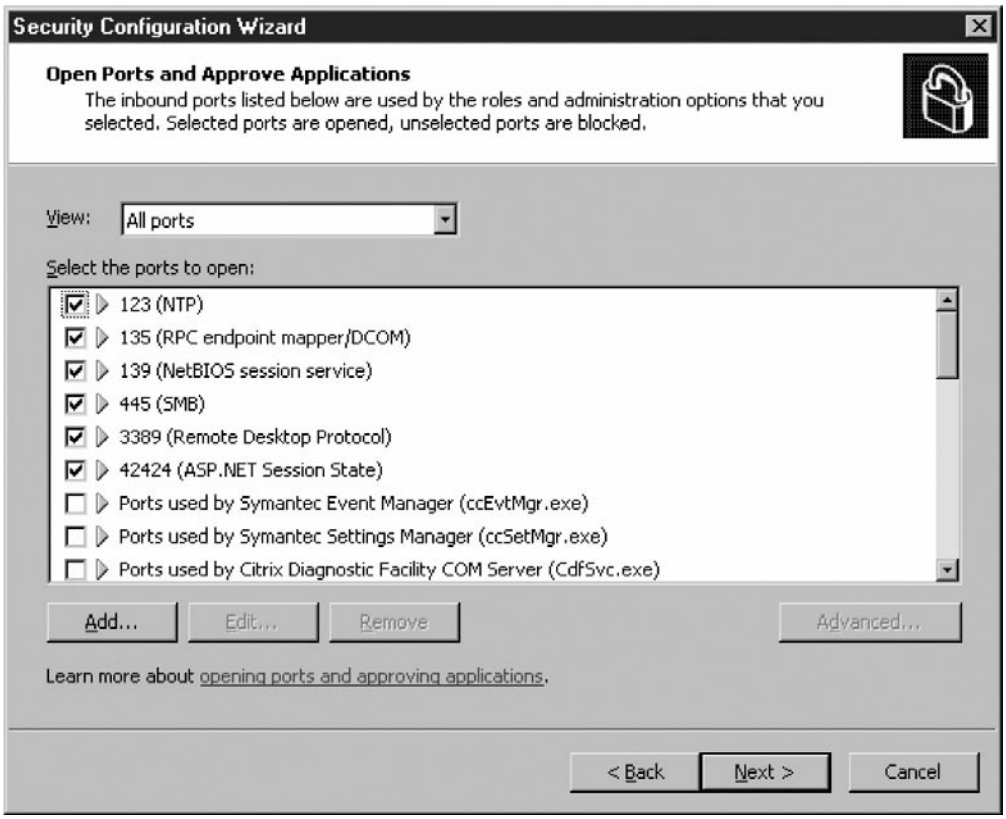
Figure 4.15 Segregating XenApp Assets



Earlier we discussed the use of security templates. The next step after configuring your system with a standardized security template is to use the new feature available in Windows 2003 Server SP1 called the *Security Configuration Wizard*, shown in Figure 4.16. This wizard is very easy to use and through a series of questions, you can quickly and easily disable unnecessary services, remove unwanted IIS virtual folders, block unused ports, configure audit settings, and lock down access to critical system files.

The Security Configuration Wizard (SCW) classifies servers based on roles. It will analyze your system and, based on the running services it finds, it will assign the server its appropriate role, like Web server or domain controller. SCW disables any services and ports not specifically tied to a defined role, which reduces the server's attack surface. Before running the wizard, you must first make sure that all applications are running correctly and are properly configured or you could inadvertently disable a needed service or port. The tool is quite granular in what you are allowed to select and you should make every effort to utilize this tool in the maintenance of your environment's security posture.

Figure 4.16 Using the Security Configuration Wizard



Understanding Client Devices

ICA Clients exchange information between a user's client device and the published application resources on XenApp. There are several versions of the Citrix client for a variety of platforms.

Your organizational policies should dictate what versions of the client software are acceptable, what devices on which they are permitted to run, and the specific configuration of the client software.

Something seemingly benign as a minor software version could potentially be used to an attacker's advantage. For example, *Citrix Security Bulletin CTX116227* states that, "Under some circumstances, the Citrix Presentation Server Client for Windows may leave residual credential information in the client process memory. This issue is present in all versions of the Citrix Presentation Server Client for Windows prior to version 10.200." Is this a major problem that is going to provide a likely avenue for an attacker to exploit? The real question is, do you want to give an attacker that opportunity regardless of how likely it is to happen?

ICA Client software is available for a range of different devices and platforms. Why should what kind of device or platform be of concern? Assume that a Citrix client is installed on a device for which your organization has not published any security lockdown policies, such as a *personal digital assistant (PDA)* that is running the Windows CE operating system. One of your users then connects to your corporate network via a public wireless link on a system that you have not locked down. The risk here is quite evident.

Different Citrix client versions also support different feature sets. You should be aware of what clients support multifactor authentication, certificate revocation checking, SSL/TLS, and so on, as shown in Table 4.1.

Table 4.1 Features Supported by Client Type

Feature	FIPS 140	TLS Support	3DES	AES	Certificate Revocation Checking	Smart Card Support	Kerberos Support
Program Neighborhood (Win32), version 10.x	X	X	X	X	X	X	X
Program Neighborhood Agent (Win32), version 10.x	X	X	X	X	X	X	
Web Client (Win31), version 10.x	X	X	X	X	X	X	X
Client for Windows CE WBT, version 10.x		X	X			X	
Client for Pocket PC, version 10.x		X	X			X	
Client of Java, version 9.x		X	X	X	X		X
Client for Macintosh, version 7.0		X	X				
Client for Macintosh, version 8.x		X	X			X	X

Continued

Table 4.1 Continued. Features Supported by Client Type

Feature	FIPS 140	TLS Support	3DES	AES	Certificate Revocation Checking	Smart Card Support	Kerberos Support
Client for Win16, version 6.20							
Client for Linux Version 10.x		X	X			X	
Client for UNIX (Sun Solaris), version 8.x		X	X			X	
Client for UNIX (IBM AIX), version 6.30		X	X			X	
Client for UNIX (SGI IRIX), version 6.30							
Client for UNIX (HP–UX), version 6.30		X	X				
Client for OS/2, version 6.012							

Understanding Users Rights, Responsibilities, and Permissions

Security is not just the responsibility of you as the administrator. It is a shared responsibility between you, your organization, anyone who uses the system, and anyone who is in the same physical location where a server or a client is located. This may sound extreme, but when it comes to security, it really is everyone's responsibility. Ever wonder why we have those *acceptable use policies* or the *warning banners* that precede a user logon? These are reminders to the users that security is serious business and must constantly be monitored. All users should be allowed the privilege of accessing the computing resources to which they are entitled as long as those purposes are consistent with your organization's mission.

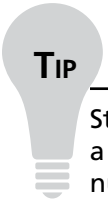
With this in mind, there are responsibilities that accompany a user's privilege. Some basic user responsibilities include:

- Protecting their accounts, passwords, and data assigned to them to the best of their ability.
- Logging out of sessions when no longer active.
- Not introducing malicious code, viruses, or other harmful elements to the system.
- Using their access only for legal purposes.
- Not sharing their account information, passwords, or access codes with others.
- Reporting violations of policy to the proper personnel.

Just as users have their responsibilities, so do the administrators. As an administrator, you are also a user and are bound by the responsibilities above. In addition, some basic administrator responsibilities regarding users include:

- Ensuring that appropriate password policies are enforced.
- Ensuring that only authorized users are allowed access to the proper information.
- Providing a reliable and productive system.
- Providing a contingency plan in the event of service interruption.
- Providing reliable data backups in the event of disaster or data loss.

Of course there are many more items that can be added to these basic lists. It is the responsibility of your organization to identify what are the most important policies, and then to ensure that they are understood and adopted by all users.



TIP

Strong passwords should be used whenever possible. A strong password should be a completely random series of characters including upper and lower case letters, numbers, special characters, and even spaces. Many systems now support *passphrases*. A passphrase is simply a sentence that the user can easily remember. Microsoft has an excellent link regarding passwords and a strong password checker at www.microsoft.com/protect/yourself/password/create.mspx.

Tools & Traps…

Passwords – The Front Door to Your Network

Most security studies have shown that passwords are by far the number one security hole on most networks. Even networks that have policies in place to enforce strong passwords may have a test environment that overrides the default password policy simply for perceived ease of use.

Password crackers are easily attainable by a potential attacker. Tools such as *Cain and Able*, *John the Ripper*, *THC Hydra*, and *L0phtcrack* are just a few. As an administrator, you should make it a part of your routine security maintenance to run password cracking tools on your own environment to see what an attacker could potentially see. How many service accounts and passwords do you have in your network? Do you maintain separate accounts and passwords for Structured Query Language (SQL), e-mail services, and Web services? Right now, do you have a Windows Web server that has a configured IUSR account? When was the last time you changed that password? Give your network a password strength test. Do all of your passwords stand up to the latest brute force password cracking tools available? The results may surprise you.

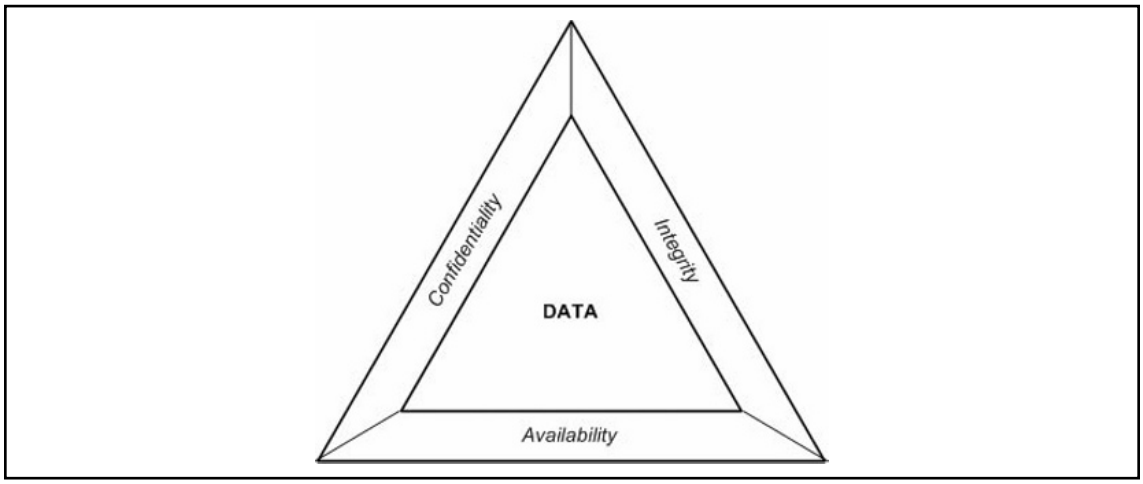
Defining Types of Deployments

When designing a XenApp environment, the operational needs of your organization have to be weighed against the security requirements of your organization. Typically, deployments should be designed in such a way that they satisfy the requirements of the *CIA triad*, shown in Figure 4.17, dealing with information security. The CIA triad stands for confidentiality, integrity, and availability. It is a very good idea when designing a XenApp deployment that you remember these three concepts:

- **Confidentiality** Confidential or sensitive information must only be accessed, used, copied, or disclosed by personnel with the proper authorization and when they have a genuine need.
- **Integrity** This insures that data cannot by created, changed, copied, deleted, viewed or processed without proper authorization.
- **Availability** This means that the data, the systems used to process the data, and controls used to protect the data are all available and functioning when the data is needed.

Following these principles, any deployment you design should be a *secure deployment* where the threats and risks to your organization's information infrastructure have either been mitigated, eliminated, or accepted.

Figure 4.17 Defining Information Security Using the CIA Triad



Internal Network (Intranet) Deployment Using SSL Relay

If your XenApp implementation does not require any external connectivity and will only be used by your internal users, you can opt to implement a standard XenApp deployment without the further need to encrypt your connections. However, if your organization's security requirements dictate the need to protect sensitive data from end to end, you will want to use the Internal Network (Intranet) Deployment using SSL Relay.

This deployment provides end-to-end encryption of the communication between the client and the server. Both SSL Relay and the appropriate server certificate must be installed and configured on each server within the server farm.

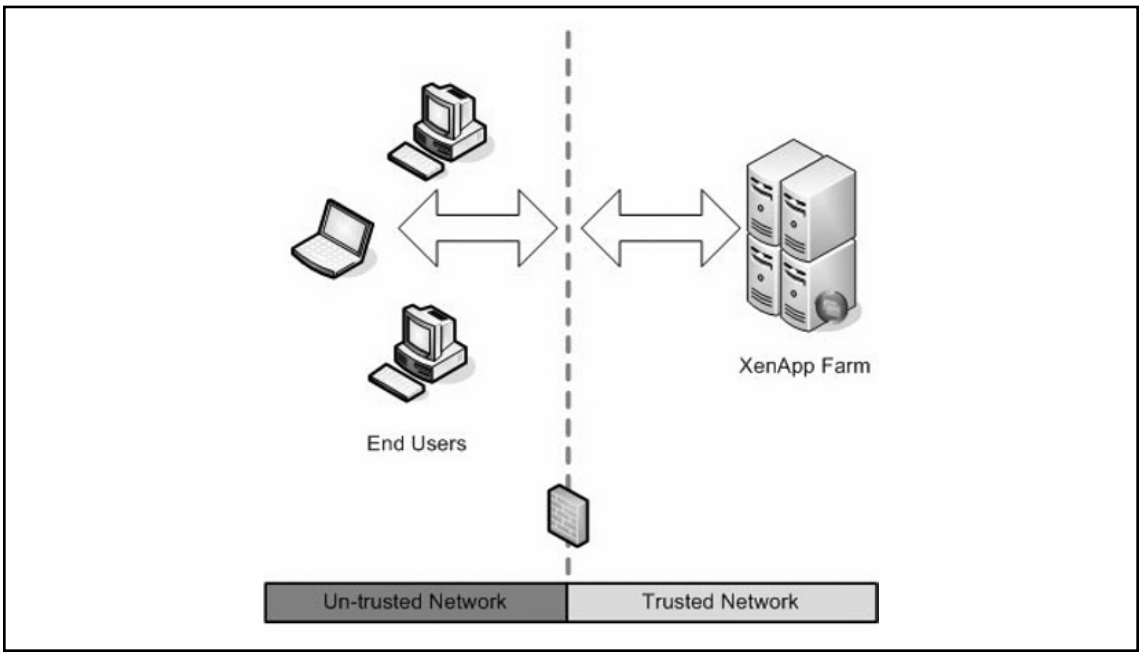
NOTE

If your public key infrastructure (PKI) employs the use of a third-party or intermediate certificate authority (CA), you will need to ensure that the appropriate CA root certificates are installed on each XenApp server and component utilizing SSL and every client that will initiate a session to your XenApp farm.

The SSL Relay operates as an intermediary component in the communications between the client and the XML Service on each XenApp server. The client authenticates the SSL Relay by checking the SSL Relay's server certificate against a list of trusted certificate authorities. After this authentication, the client and SSL Relay negotiate requests in encrypted form. The SSL Relay decrypts the requests and forwards the requests to the XenApp server. When returning the information to the client, the XenApp server sends all information through the SSL Relay, which encrypts the data and forwards it to the client to be decrypted.

Depending on your network configuration, there may or may not be a firewall involved in this deployment. In Figure 4.18, we have shown the deployment utilizing a firewall. Either with or without a firewall, the only traffic between client and server that is utilized is TCP 443. A comprehensive diagram of port traffic flow for this configuration is provided in Figure 4.22.

Figure 4.18 Internal Network Deployment Using SSL Relay



External Network Deployment (Single Hop)

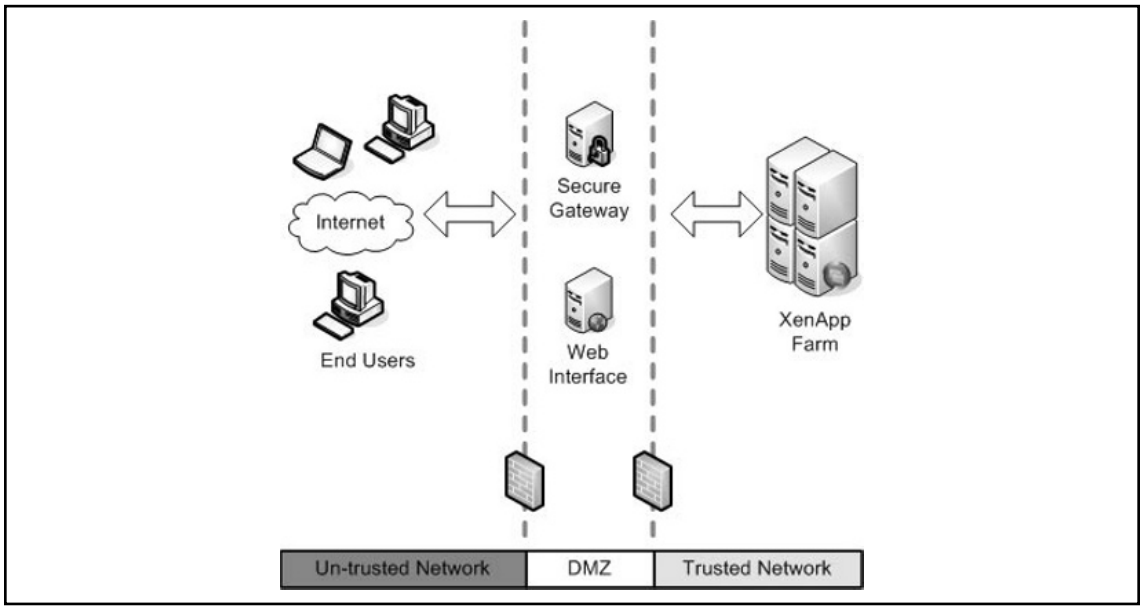
If your XenApp implementation requires any sort of external connectivity, and you've just lost the fight with the firewall team asking them to punch a hole in the firewall to allow ICA traffic (TCP 1494) through, then you will need to look at the External Network Deployment (Single Hop). This deployment's data traffic flow goes from the XenApp server to the DMZ to the client, thus providing a single hop.

This deployment requires the use of a Web Interface server to be installed on a server running Internet Information Services (IIS) and a Secure Gateway server. Note that IIS does not need to be installed on the Secure Gateway server. A typical configuration of this type of deployment utilizes a firewall between the XenApp farm and the DMZ and a firewall between the DMZ and the client. In this type of deployment, it is possible to install the Web Interface and the Secure Gateway on a single machine; however, this configuration is not recommended for most environments.

The deployment in Figure 4.19 outlines the *parallel configuration* of the Single Hop deployment which utilizes a single DMZ. This configuration exposes both the Secure Gateway and the Web Interface to the Internet and requires that both servers have connectivity available to clients. The *inline configuration* of the Single Hop deployment places the Secure Gateway in front of the Web Interface causing only the Secure Gateway to be exposed to the Internet. However, by using the inline configuration of the Single Hop deployment, you are limited in your authentication methods. For example, when the Secure Gateway is placed in front of the Web Interface, smart card authentication is not supported.

Utilizing the parallel deployment provides for more authentication methods than are available utilizing the inline deployment. Figure 4.20 shows port traffic utilizing this configuration. You may further secure this configuration by configuring SSL Relay between the Web Interface and the XML service running on a XenApp server. To achieve FIPS 140 compliancy, you can secure the communication between the Secure Gateway and Citrix XenApp Server using IPsec policies.

Figure 4.19 External Network Deployment (Single Hop)



External Network Deployment (Double Hop)

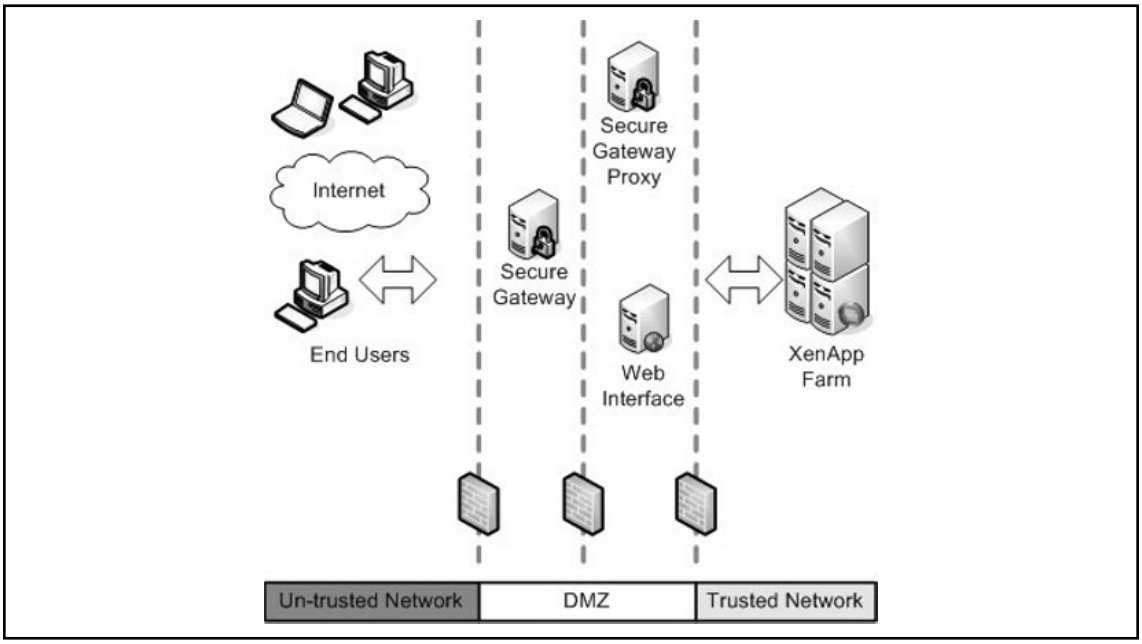
The External Network Deployment (Double Hop) is shown in Figure 4.20 and makes use of two DMZs. Only the Secure Gateway is exposed to the Internet. If you choose not to expose your Web Interface to the Internet or your network configuration mandates the use of two DMZs, then this deployment is appropriate for your environment.

In this deployment the DMZ is divided into two segments or zones. The Secure Gateway is placed in the outward facing segment and the Web Interface and the Secure Gateway Proxy are placed in the second segment (or second hop).

This deployment requires the use of a Web Interface server to be installed on a server running Internet Information Services (IIS), a Secure Gateway server, and a Secure Gateway Proxy server. Note that IIS does not need to be installed on the Secure Gateway server or the Secure Gateway Proxy server. A typical configuration of this type of deployment utilizes a firewall between the XenApp farm and the first DMZ, a firewall between the first DMZ and the second DMZ and a firewall between the second DMZ and the client.

This configuration exposes only the Secure Gateway to the Internet. However, by using this deployment, you will not be able to utilize Smart Card authentication. Figure 4.24 provides a detailed examination of the ports used and their flow in this deployment. You may further secure this configuration by configuring SSL Relay between the Web Interface and the XML service running on a XenApp server. To achieve FIPS 140 compliancy, you can secure the communication between the Secure Gateway and Citrix XenApp Server using IPSec policies.

Figure 4.20 External Network Deployment (Double Hop)



Web Interface with SSL Relay

If your XenApp implementation does not require any external connectivity and will only be used by your internal users, you can opt to implement a standard XenApp deployment without the further need to encrypt your connections. However, if your organization's security requirements dictate the need to protect sensitive data from end to end, you will want to use the Internal Network (Intranet) Deployment using SSL Relay.

This deployment provides end–to–end encryption of the communication between the client and the server. Both SSL Relay and the appropriate server certificate must be installed and configured on each server within the server farm.

The SSL Relay operates as an intermediary component in the communications between the client and the XML Service on each XenApp server. The client authenticates the SSL Relay by checking the SSL Relay's server certificate against a list of trusted certificate authorities. After this authentication, the client and SSL Relay negotiate requests in encrypted form. The SSL Relay decrypts the requests and forwards the requests to the XenApp server. When returning the information to the client, the XenApp server sends all information through the SSL Relay, which encrypts the data and forwards it to the client to be decrypted.

Depending on your network configuration, there may or may not be a firewall involved in this deployment. In Figure 4.21, we have shown the deployment utilizing a firewall. Either with or without a firewall, the only traffic between client and server that is utilized is TCP 443 (see Figures 4.22 to 4.24). A comprehensive diagram of port traffic flow for this configuration is provided in Figure 4.25.

Figure 4.21 Web Interface Using SSL Relay

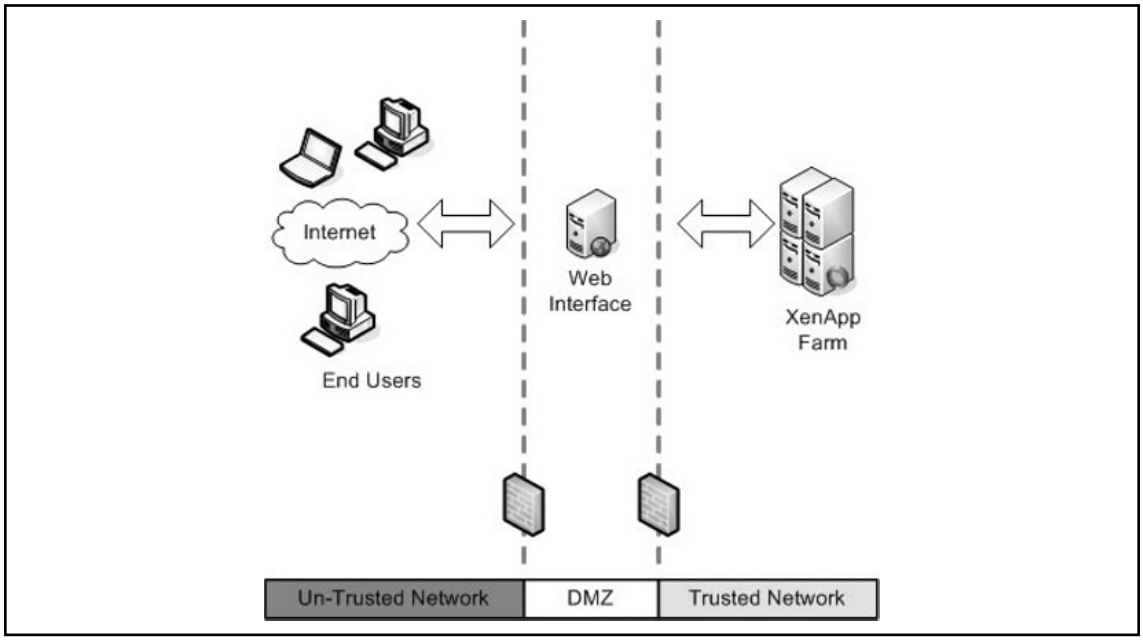


Figure 4.22 Understanding the SSL Relay Configuration

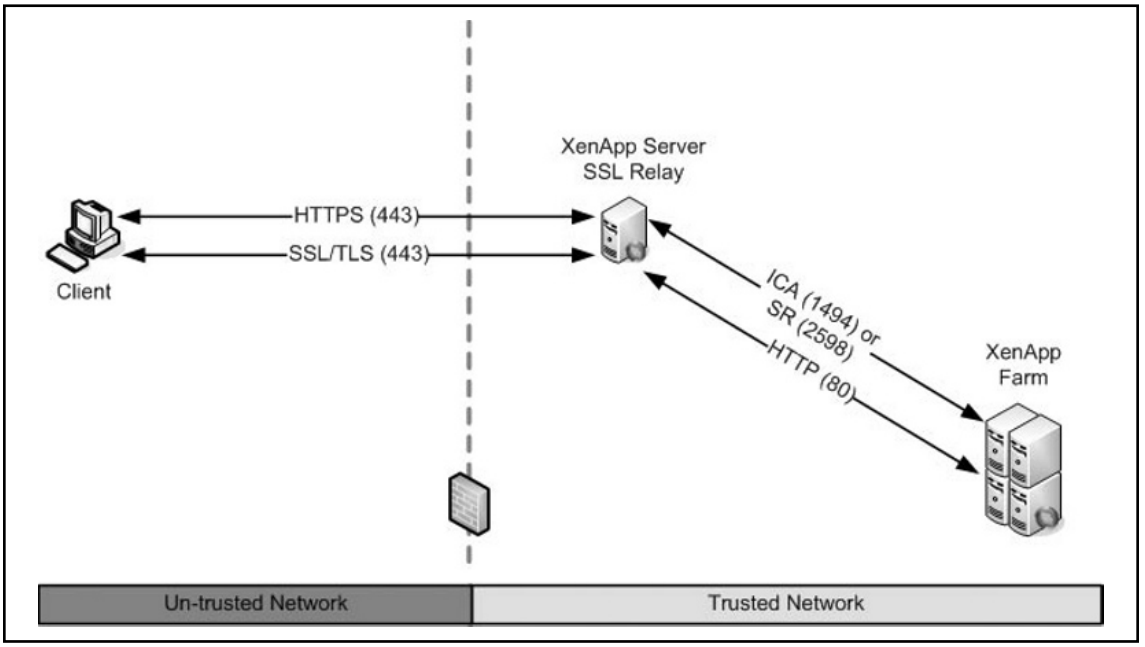


Figure 4.23 SSL Relay through a Web Interface and Secure Gateway

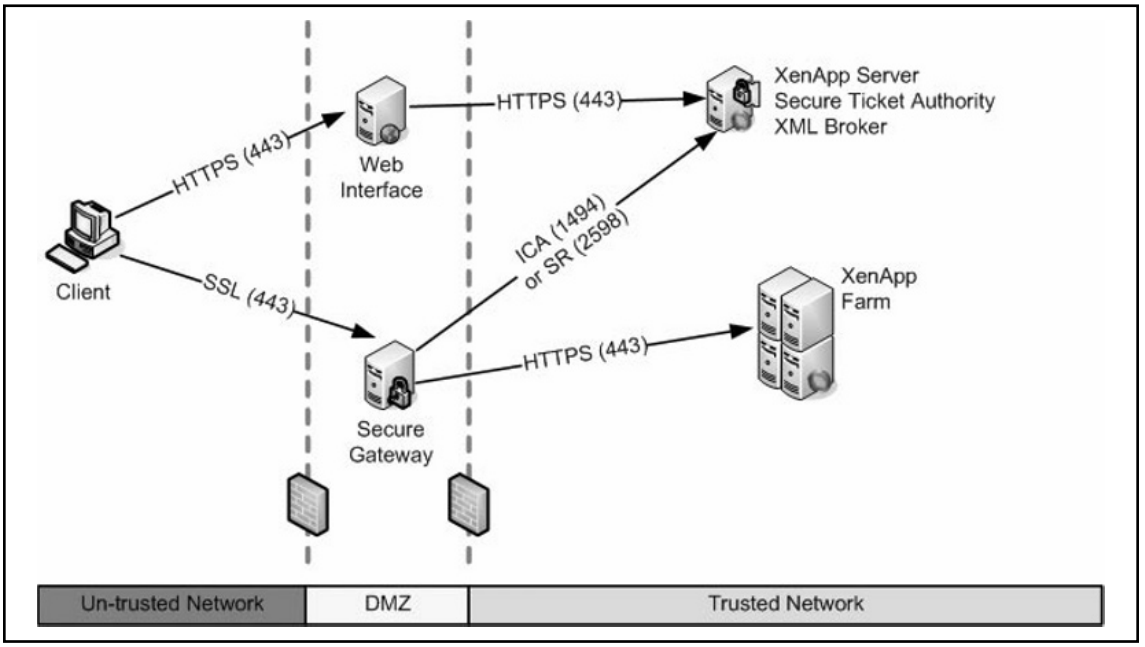


Figure 4.24 Understanding the Double–Hop Configuration

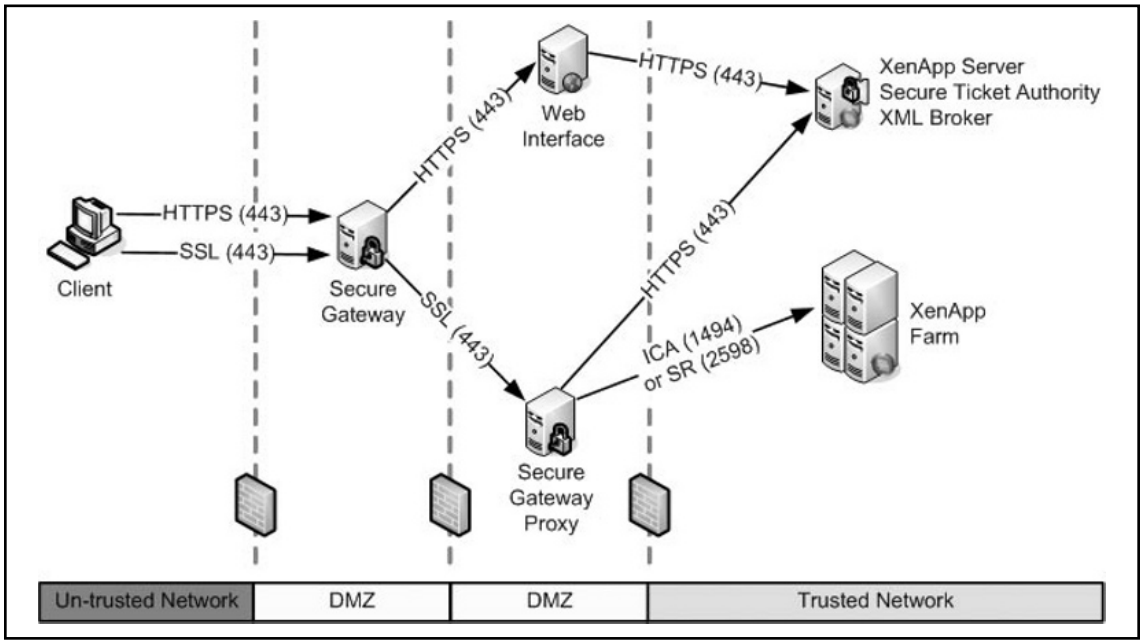
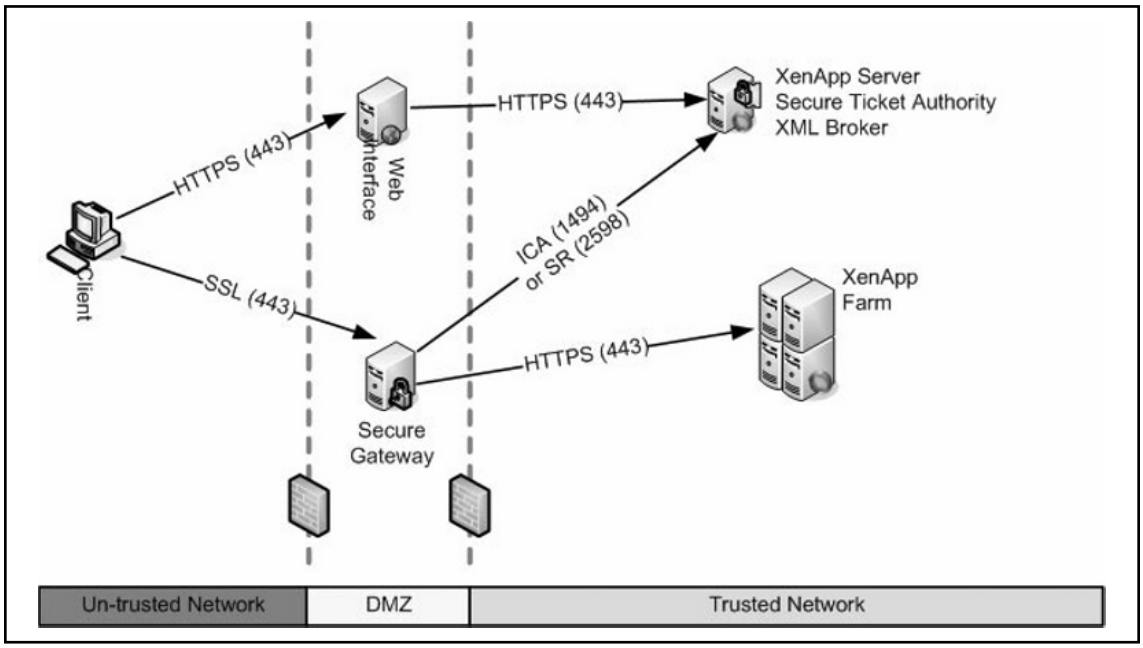


Figure 4.25 Understanding the SSL Relay Configuration with Web Interface



Understanding Wireless LANs (WLANs)

The mobile Internet is a trend that is now a common part of the workplace. People want to access the Internet from any location, using any device and without needing any wires. Once users are connected to the Internet, it is not a far leap to want to access and run applications. Now with the user of broadband wireless internet and the increased memory and processing power of PDAs, users can essentially have a usable workstation right in their pocket.

As the world continues to move further through the information age, businesses are turning more to technology to gain a competitive advantage in their respective industries. Every day, businesses make strategic decisions based on information that is as up to date as they can obtain. It follows that the ability of an organization to send and receive information and thus act on that information faster than its competitors gives it a distinct advantage over competition. Until now, no one has been able to collect and act on information in a real-time manner.

One of the biggest concerns facing network administrators in implementing a WLAN is data security. In a wired environment, the lack of access to the physical wire can prevent someone from wandering into your building and connecting to your internal network. In a WLAN scenario, it is impossible for the AP to know if the person operating the wireless device is sitting inside your building, passing time in your lobby, or seated in a parked car just outside your office.

Understanding Authentication Methods

Citrix XenApp provides a variety of authentication methods to use. Explicit authentication using the traditional user name and password combination where the user is prompted to enter their information, pass-through authentication where the existing log-on credentials of the current logged on session are passed to the XenApp session, authentication methods utilizing Kerberos, Smart cards, biometrics

Understanding Explicit Authentication

Explicit authentication is simply forcing the user to enter their userid and password. Explicit authentication is an option on XenApp clients and the Web Interface.

Understanding Kerberos Authentication

Kerberos, version 5, is an industry standard security protocol that Windows Server 2003 uses as the default authentication service. It is used to handle authentication in Windows Server 2003 trust relationships, and is the primary security protocol for authentication within domains. Kerberos uses mutual authentication to verify the identity of a user or computer, and the network service being accessed. Each side proves to the other that they are who they claim to be. Kerberos does this through the use of tickets.

Kerberos authentication can only be used by network clients and servers running Windows 2000, Windows Server 2003, or Windows XP Professional; any Windows 9x or NT clients that attempt to access a Kerberos secured resource will use NTLM authentication instead.

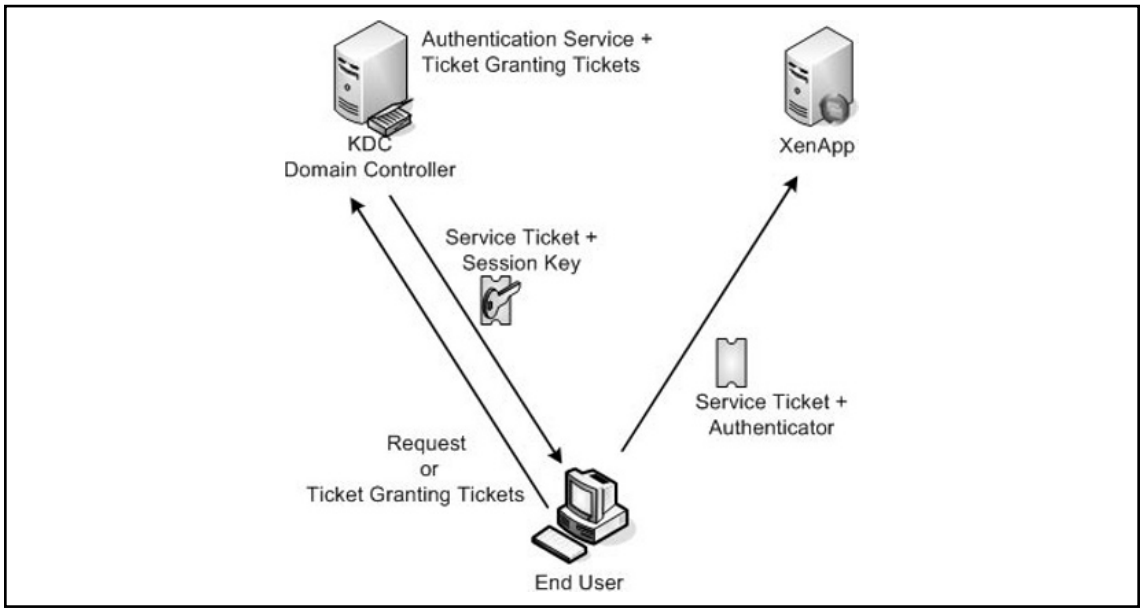
Kerberos authentication relies on a Key Distribution Center (KDC) to issue tickets to enable client access to specific network resources. Each domain controller (DC) in a Windows Server 2003 domain functions as a KDC, which creates fault tolerance in the event that a DC becomes unavailable.

Network clients will use Domain Name Service (DNS) to locate the nearest available KDC; once they've located the KDC they will provide a passphrase in order to acquire a ticket. Kerberos tickets contain an encrypted password that confirms the user's identity to the requested service. These tickets will remain active on a client computer system for a configurable amount of time, usually 8 or 10 hours. The longevity of these tickets allows Kerberos to provide single sign-on capabilities, where the authentication process as a whole becomes transparent to the users once they've initially entered their log-on credentials.

These steps occur completely behind the scenes; the users are only aware that they've entered their password or Personal Identification Number (PIN) number as part of a normal log-on process. The Kerberos process is shown in Figure 4.26.

1. Using a smart card or a username/password combination, a user authenticates to the KDC. The KDC issues a ticket-granting ticket (TGT) to the client system. The client retains this TGT in memory until needed.
2. When the client attempts to access a network resource, it presents its TGT to the ticket-granting service (TGS) on the nearest available Windows Server 2003 KDC.
3. If the user is authorized to access the service that it is requesting, the TGS issues a service ticket to the client.
4. The client presents the service ticket to the requested network service. Through mutual authentication, the service ticket will prove the identity of the user and the identity of the service.

Figure 4.26 Understanding How Kerberos Works



WARNING

Kerberos authentication relies on timestamps to function properly. As such, all clients that are running the Kerberos client must synchronize their time settings with a common time server. If the time on a network client is more than five minutes slow or fast compared to the KDC, Kerberos authentication will fail.

Kerberos is supported in the XenApp environment, but there are specific requirements that must be met before it will work successfully. Requirements for Kerberos Support with XenApp:

- XenApp on Windows 2003
- ICA Client, version 8.x or higher
- Client\Server connections must be in same domain or have trusts between domains.
- Servers must be "**trusted for delegation**" (must be configured through **AD Users and Computers management tool**).
- SSPI requires the XML Service, DNS service address resolution to be enabled for the server farm, or reverse DNS resolution to be enabled for an AD domain.

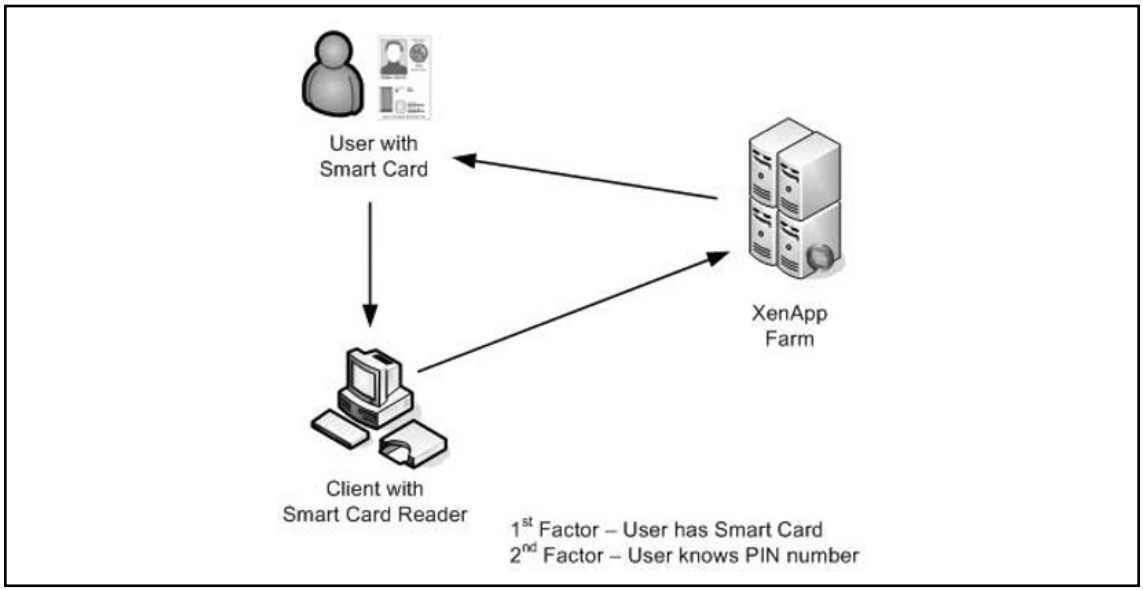
Because of the security complexities required of many XenApp environments, even though you may want to implement a Kerberos authentication scheme, you may not be able to do so because of your specific configuration. Kerberos will not work in any of the following conditions:

- Within your terminal services configuration if you have **Use standard Windows authentication** enabled, or **Always use the following log-on information** is completed, or the **Always prompt for password** option is checked
- Your configuration utilized connections through the Secure Gateway
- If the XenApp server is configured to *require* smart card logon.
- If the user account requires a *smart card* for interactive logon.

Understanding Multifactor Authentication

An authentication factor is a piece of information and process used to authenticate a person's identity for security purposes. Two-factor authentication (2FA), shown in Figure 4.27, is an authentication mechanism based on two pieces of information: something you have, such as a smart card, token id, etc. and something you know, such as a PIN. When presented with a log-on option, the user must provide both pieces of the authentication mechanism or they will be denied access to the system. There is another factor of authentication providing multifactor authentication based on something a person is or does, such as a biometric recognition. Citrix XenApp natively supports 2FA and can support 3FA with third-party add-on products.

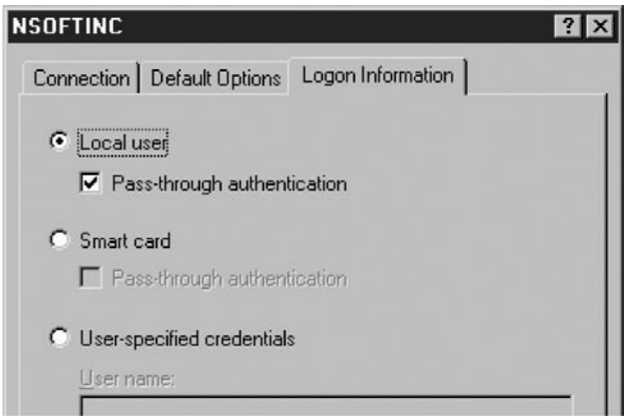
Figure 4.27 Understanding Multifactor Authentication



Understanding Pass-Through Authentication

Pass-through authentication allows the user's name and password to be passed from the local machine to the server. If you do not elect to install Pass–through authentication during the initial installation of XenApp server and decide that you want the feature later on, you will need to reinstall the pass-through client. However, the use of pass-through credentials may be contrary to your organization's security policies unless pass–through authentication is used in conjunction with other technologies, such as smart cards, that can provide multifactor authentication. Pass-through authentication can be enabled on the Citrix client as shown in Figure 4.28.

Figure 4.28 Enabling Pass-Through Authentication



Encrypting XenApp

In addition to the numerous security solutions and products available on the market today, XenApp provides built-in capability to help secure server and client communications from intruders. Using a standard technology known as *encryption*, server-to-server and client/server communication can be protected against intruders. Understanding how encryption works and where to apply it is important to ensure proper implementation of a secure server farm. Once you understand how encryption is used, you can properly set up the products and add-ons provided for XenApp.

Understanding Encryption

Encryption is the process of converting data into nonreadable text, also referred to as *ciphertext*. Ciphertext is used for transmitting confidential data. Once the data has arrived at its destination, it is then reconverted into the original data through a process known as *decryption*.

Various types of data encryption are available on the market today. Each type provides both benefits and disadvantages, including categories such as strength of security, ease of use, and standardization. The effectiveness of any security algorithm is found in its strength and the keys used to secure it. Weaker security algorithms are more easily cracked; however, if implemented properly, these options can still provide very effective solutions.

Encryption techniques are based on using keys similar to keys for your home or car. Using a key to open your car door is a method that identifies you as someone authorized to use the car. Whoever has possession of this key is able to access the car. Based on mathematical algorithms, encryption keys work the same way. If you have the correct key, you can encrypt or decrypt the data from ciphertext.

Symmetric Key Encryption

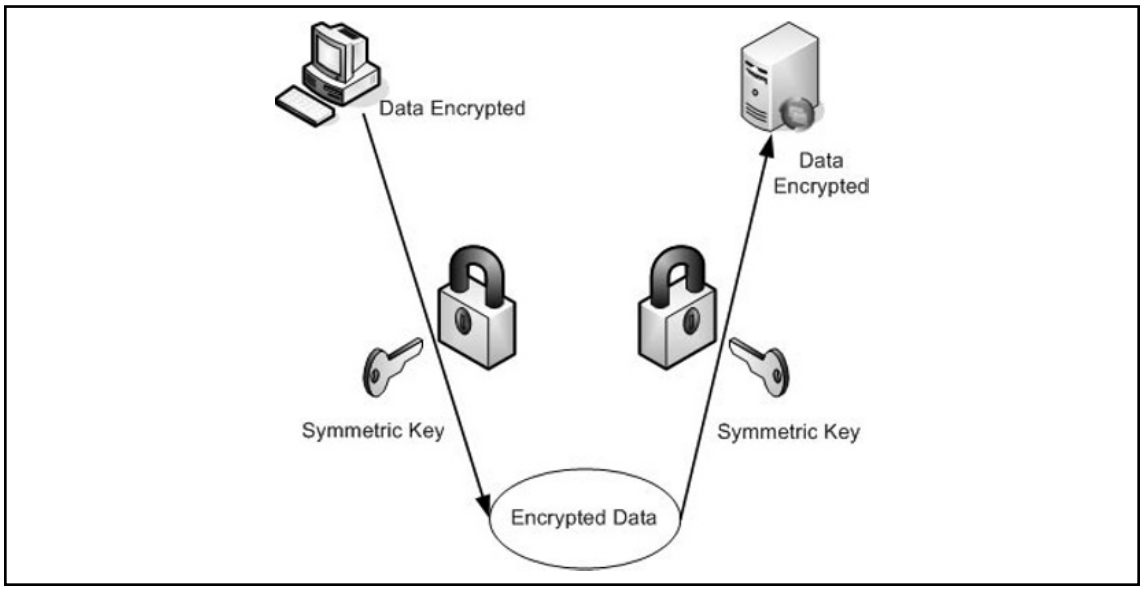
Primarily two forms of encryption are in use today. The most common form of encryption uses the *symmetric algorithm*. This method requires that each individual or device that accesses this encrypted data possess a copy of the key. Commonly referred to as *shared-key encryption* because it uses a single key for encryption and decryption, this is a relatively simple encryption technology to implement, but it might not provide the best security.

One common issue with symmetric encryption algorithms is the way the keys are transported to other users. If this type of key is obtained by an unauthorized user (such as during the encryption setup process), that unauthorized user can then easily decrypt that data, resulting in loss of data integrity. Most solutions that use symmetric keys to encrypt data also provide additional secure methods by which to negotiate and transport these keys to ensure that they are secured.

Another issue associated with symmetric keys is managing multiple identities. If you want to communicate securely using symmetric keys without all users having access to all data, you must maintain different keys for different data sets. For example, Jane at Company A wants to send data to Bill at Company B. Jane must configure communications to Bill using Key 1, and Bill must use the same key. Jane also wants to communicate with Bob at Company X. To communicate with Bob, Jane must use a different key; therefore, Key 2 is created. As the number of companies or individuals grows, so does the number of keys required to maintain communications among them. Now imagine using this technology on an enterprise scale with hundreds or thousands of sites.

A common symmetric algorithm implementation used today is the Digital Encryption Standard, or DES. Based on a fixed 56-bit symmetric key, this algorithm creates a single key based on a binary number used to encrypt and decrypt data. Using a block cipher methodology, it uses 64-byte blocks to randomly populate a key. Currently in use by organizations such as the National Security Agency (NSA), DES offers 72 quadrillion possible encryption keys at this point. In addition, developers have created a stronger version of DES, known as Triple DES, or 3DES, because it uses the DES key by encrypting, decrypting, and encrypting again to ensure data is secure. Figure 4.29 shows an example of symmetric key processing.

Figure 4.29 Symmetric Key Encryption



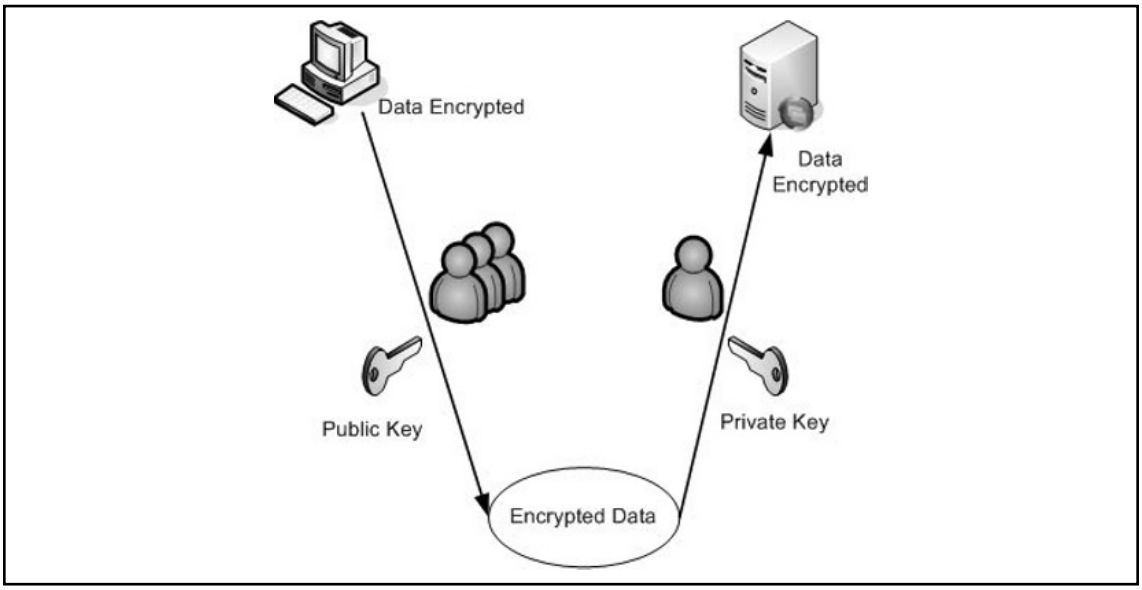
Asymmetric Key Encryption

The second method of encryption uses an asymmetrical algorithm and is commonly known as *public key encryption*. Although similar to the symmetric algorithm, this technology uses two keys to encrypt data. The first key is held privately in a secure location for the receiver to decrypt data sent to him. This validates that the receiver is authentic. The second key is freely published, is used to encrypt the data, and is commonly posted in public locations. This allows anyone to send the data, but only the holder of the private key is authorized to receive and decrypt the data. Even the public key originally used to encrypt the message cannot be used to decrypt it. This encryption technique allows you to send the public key over insecure channels and still maintain the integrity of encrypted data.

Therefore, if Tom wants to send encrypted data to Jim, he uses Jim's public key to encrypt it. Once the data is transferred to Jim, he uses his private key to decrypt the data. This methodology ensures that only Jim can access the data. Created by and named for Ron Rivest, Adi Shamir, and Leonard Adleman, the Rivest–Shamir–Adleman (RSA) data encryption standard is the most commonly used asymmetric algorithm. It uses prime numbers to randomly generate public and private keys. A common application

using RSA encryption includes Pretty Good Privacy (PGP) and Novell NetWare for a secure client-to-server communications channel. Similar to RSA, Diffie-Hellman is another common algorithm. Primarily used to transfer symmetric keys securely, Diffie–Hellman provides another form of asymmetric keys. A common example of asymmetric key encryption is using PGP to digitally sign e-mail communications. If you use a PGP signature, recipients of your messages can ensure that your e-mails are authentic. Figure 4.30 shows an example of how asymmetric keys work.

Figure 4.30 Asymmetric Key Encryption



NOTE

When you are implementing encryption within your XenApp farm, it is critical to correctly identify the encryption strength to use and where it will be configured. All encryption strengths, with the exception of basic, use 128-bit strength for the log-on process. Afterward, they revert to the selected option, such as 56-bit. You might ask, "Why not just use 128-bit encryption if it's the best?" Unfortunately, the stronger the encryption algorithm, the more performance overhead is required on the server and client communications. For example, 128-bit encryption will not function properly when you use a 33.6Kbps modem connection to access a XenApp. In the same manner, client performance degrades faster over inconsistent network links when you use higher strength encryption algorithms. Carefully test each option to ensure it suits your environment; it can have a major impact on the performance of your server farm if not optimized properly.

Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) was created to encrypt data transmitted between a client computer and a Web server. Traditionally, Web traffic is transmitted in cleartext, potentially providing network intruders with sensitive data. Netscape developed SSL to provide a secure communications method by which to converse across the Internet. Based on RSA public/private key technology using digital certificates, SSL has become the standard for secure communication across the World Wide Web and can be used to complement your security strategy for your XenApp farm.

SSL is classified as a transport layer security protocol, since it secures not only the information generated at the application layer, but at the transport layer as well. It is considered a secure protocol by providing the mechanisms for supporting the basic elements of secure communications, namely confidentiality, integrity, and authentication.

Authentication ensures that the information received is indeed from the individual believed to be the sender. Integrity guarantees that the message received is the same message that was sent, while confidentiality protects data from inspection by unintended recipients.

SSL protects information passed by application protocols such as File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), and Network News Transfer Protocol (NNTP). An application must be explicitly designed to support SSL's security features. Unlike Layer 3 protocols, it is not transparent to application layer processes.

SSL uses several protocols to provide security and reliable communications between client and server SSL-enabled applications. Specifically, the handshake protocol negotiates levels and types of encryption, and sets up the secure session. These include SSL protocol version (2.0 or 3.0), authentication algorithms, encryption algorithms, and the method used to generate a shared secret or session key.

SSL uses a record protocol to exchange the actual data. A shared session key encrypts data passing between SSL applications. The data is decrypted on the receiving end by the same shared session key. Data integrity and authentication mechanisms are employed to ensure that accurate data is sent to, and received by, legitimate parties to the conversation. SSL uses an alert protocol to convey information about error conditions during the conversation. It is also used by SSL hosts to terminate a session.

SSL is used to confirm the identity of a server or a client machine and then encrypt all traffic between the two devices. For example, when you process a credit card transaction through a Web site, you want to ensure the identity of the receiver. SSL allows digital signatures to be used and are verified by a trusted certificate authority (CA). When you connect to a Web site using SSL, a certificate is processed, validating that the Web site is authentic. If it isn't, an error is issued, allowing you to determine whether to continue. As the process is completed, all traffic between your client and the Web site is encrypted to ensure that someone else cannot monitor the data flow.

SSL can be very useful when you're trying to secure a Web Interface server. Using SSL allows you to first confirm that your Web site is authentic to users. In addition, traffic such as authentication will not be sent in cleartext, increasing your security risk. To use SSL with Web sites, you connect using Secure Hypertext Transfer Protocol (HTTPS) instead of HTTP. Once this is configured, you no longer have to use standard HTTP services—you can rely solely on HTTPS to ensure that your site is secured.

How a Secure Channel is Established

To understand how a secure channel is formed, let's examine how an SSL client establishes a session with an SSL Web server:

1. A URL is entered into a Web browser using HTTPS rather than HTTP as the protocol. SSL uses TCP Port 443 rather than Port 80. The HTTPS entry requests the client to access the correct port on the target SSL Web server.
2. The SSL client sends a client Hello message. This message contains information about the encryption protocols it supports, what version of SSL it is using, what key lengths it supports, what hashing algorithms to use, and what key exchange mechanisms it supports. The SSL client also sends a challenge message to the SSL server. The challenge message will later confirm the identity of the SSL-enabled server.
3. The server then sends the client a Hello message. After examining methods supported by the client, the server returns to the client a list of mutually supported encryption methods, hash algorithms, key lengths, and key exchange mechanisms. The client will use the values returned by the server. The server also sends its public key, which has been signed by a mutually trusted authority (a digital certificate of authenticity).
4. The client then verifies the certificate sent by the server. After verifying the server certificate, the client sends a master key message. The message includes a list of security methodologies employed by the client and the session key. The session key is encrypted with the server's public key (which the server sent earlier in the server Hello message).
5. The client sends a client finished message indicating that all communications from this point forward are secure.

Almost all messages to this point have been sent in cleartext, implying that anyone listening in on the conversation would be able to read all parts of the exchange. This is not a problem, since no information other than the session key is secret. Moreover, the session key is safe because it is encrypted with the server's public key. Only the server is able to decrypt the session key by using its private key. The next series of events takes place in a secure context.

6. The server sends a server verify message to the SSL client. This message verifies that the server is indeed the server with which the client wishes to communicate. The server verify message contains the challenge message the client sent earlier in the conversation. The server encrypts the challenge message with the session key. Only the legitimate server has access to the session key. When the client decrypts the challenge message encrypted with the session key, and it matches that sent in the challenge, then the server has verified itself as the legitimate partner in the communication.
7. The last message used to set up the secure SSL channel is the server finish message. The SSL server sends this message to the SSL client informing of its readiness to participate in data transmission using the shared session key. The SSL session setup is complete, and data passes through a secure SSL channel.

The setup procedure is dependent on several security technologies, including public key encryption, symmetric encryption, asymmetric encryption, message hashing, and certificates. In the following sections, we define these terms and see how SSL uses them to create a secure channel.

Transport Layer Security (TLS)

Transport Layer Security (TLS) is the latest, standardized version of the SSL protocol. TLS is an open standard and like SSL, TLS provides server authentication, encryption of the data stream, and message

integrity checks. Although their differences are minor, TLS 1.0 and SSL 3.0 are not interchangeable. If the same protocol is not supported by both parties, the parties must negotiate a common protocol to communicate successfully. The primary goal of the TLS Protocol is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the *TLS Record Protocol* and the *TLS Handshake Protocol*. At the lowest level, layered on top of some reliable transport protocol, like TCP, is the TLS Record Protocol. The TLS Record Protocol provides connection security that has two basic properties: privacy and reliability.

The TLS Record Protocol is used for encapsulation of various higher level protocols. One such encapsulated protocol, the TLS Handshake Protocol, allows the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before the application protocol transmits or receives its first byte of data. The TLS Handshake Protocol provides connection security that has three basic properties:

- The peer's identity can be authenticated using asymmetric, or public key cryptography.
- The negotiation of a shared secret is secure: the negotiated secret is unavailable to eavesdroppers, and for any authenticated connection the secret cannot be obtained, even by an attacker who can place himself in the middle of the connection.
- The negotiation is reliable: no attacker can modify the negotiation communication without being detected by the parties to the communication.

One advantage of TLS is that it is application protocol independent. Higher level protocols can layer on top of the TLS Protocol transparently. The TLS standard, however, does not specify how protocols add security with TLS; the decisions on how to initiate TLS handshaking and how to interpret the authentication certificates exchanged are left up to the judgment of the designers and implementors of protocols which run on top of TLS.

Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) is a Federal Information Processing Standard (FIPS), specifically FIPS Publication 197, that specifies a cryptographic algorithm that can be used to protect electronic data for use by the United States Government to protect sensitive, unclassified information. The AES algorithm is a symmetric block cipher that can encrypt and decrypt information. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

In addition to the increased security that comes with larger key sizes, AES can encrypt data much faster than Triple-DES, a DES enhancement that essentially encrypts a message or document three times. It is based on the Rijndael algorithm, named for Belgian researchers Vincent Rijmen and Joan Daemen, who developed it.

NOTE

The following Citrix clients support the AES cipher for connections using TLS: Win32 10.x, Linux x86 10.x, and Java 9.x. You should check the Citrix Web site at www.citrix.com for the latest list of ICA clients that support AES and TLS. Remove hyperlink for Web site?

FIPS 140-2

Federal Information Processing Standard 140 (FIPS 140) is a U.S. federal government standard that details a benchmark for implementing cryptographic software. It provides best practices for using cryptographic algorithms, managing key elements and data buffers, and interacting with the operating system. An evaluation process that is administered by the National Institute of Standards and Technology's (NIST) National Voluntary Laboratory Accreditation Program (NVLAP) allows encryption product vendors to demonstrate the extent to which they comply with the standard, and thus, the trustworthiness of their implementation.

To facilitate implementing secure application server access and to meet the FIPS 140 requirements, XenApp products can use cryptographic modules that are FIPS 140-validated in Windows 32-bit implementation of secure SSL/TLS connections.

The following XenApp components can use cryptographic modules that are FIPS 140-validated:

- Citrix Clients for 32-bit Windows (including Program Neighborhood, Program Neighborhood Agent, and the Web Client)
- Secure Gateway
- XenApp
- Citrix SSL Relay
- Web Interface

When using the client and server components listed above with the SSL/TLS connection enabled, the cryptographic modules that are used are FIPS-140 validated.

Encryption Strength Options

Another important factor in implementing encryption strategies is defining the strength of your solution. In addition to the encryption techniques used, the key length is a major factor in determining the strength of any algorithm. For example, 16 bits in a key provide 65,536 possible key combinations. As the number of bits increases, so does the number of key variations. When you factor the computing power of today's computers, the ability to try every combination of larger keys, such as 128 bits, can take a few years to complete.

XenApp has five encryption levels from which to select:

- Basic
- 128-bit login only
- 40-bit
- 56-bit
- 128-bit

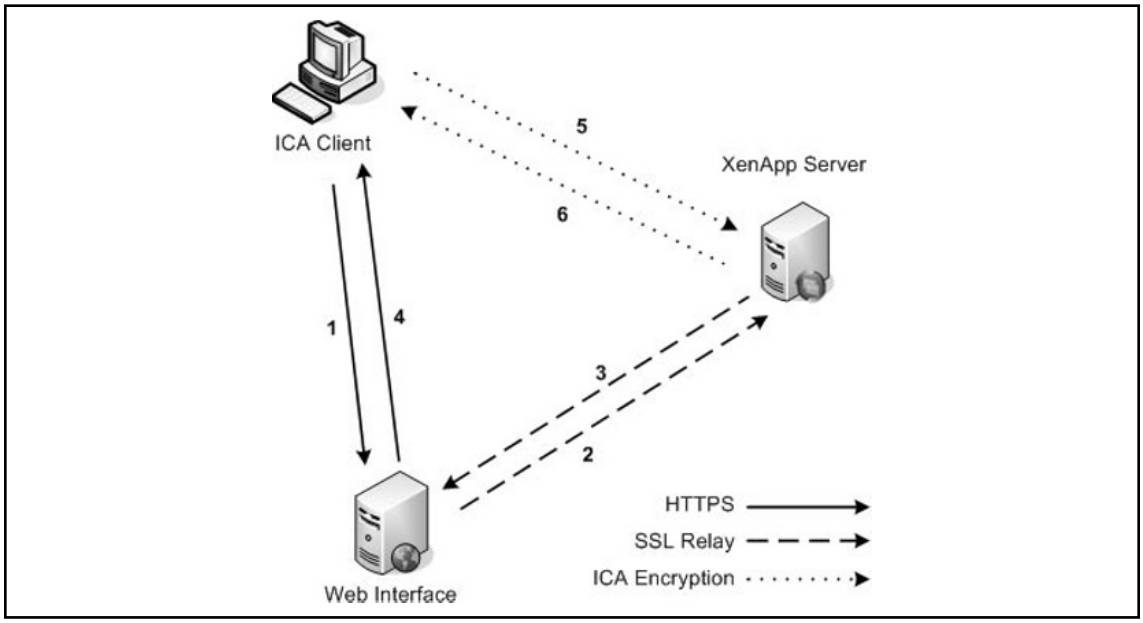
Each option provides 128-bit encryption for the log-on process, and then the selected key strength is used to secure the remainder of ICA traffic throughout the session. Once a session has been established, all traffic, with the exception of a small encryption header, will be secured. This traffic includes items such as:

- Keystrokes
- GUI information
- Mouse data
- Client drive data
- Client printer data

Where Can You Use Encryption?

Using a combination of symmetric and asymmetric key technologies, XenApp provides a comprehensive encryption solution to ensure secure communications. First, XenApp uses RC5, a fast block cipher developed for RSA security, as the symmetric key technology to encrypt all ICA traffic between clients and servers. To exchange these keys securely, XenApp has implemented the Diffie-Hellman asymmetric key algorithm. When an ICA client session is initiated, a unique public/private key pair is generated and passed through the communications channel. Once communication is established, these key pairs are used to arrive at the same RC5 symmetric key. Using a 1024-bit symmetric key, the client then begins processing ICA traffic and log-on information. Figure 4.31 illustrates how each of the encryption technologies is used to provide a complete solution. The communications path for initiating a connection via the Web Interface to a XenApp farm is shown.

Figure 4.31 Encryption Technologies at Work



1. Client connects to the Web Interface using HTTPS.
2. The Web Interface server validates the user and requests available applications using SSL Relay service to encrypt traffic to the XenApp server.

3. XenApp returns published applications available for this user.
4. The Web Interface server provides a Web page using HTTPS with available applications.
5. ICA client connects directly to the XenApp server providing published applications.
6. A communications channel is opened using ICA encryption, and the client session begins.

Encrypting Server, Published Application, and Client Communications

XenApp offers multiple encryption techniques providing flexibility in managing secure sessions. As discussed within this chapter, encryption is a key component ensuring secure communications between the XenApp farm and the ICA client. There are several ways to configure encryption:

- Encrypting traffic at the server level
- Setting encryption for each published application
- Setting encryption on an individual client basis

You must understand how each option affects the overall environment and how the options can be used together successfully. The first option is encrypting traffic for all connections coming into the server. When you select this option, all ICA sessions initiated to the configured server encrypt data to the specified strength. This option mandates that each server must be configured independently, requiring administrators to touch each server any time this setting must be modified. For larger server farms, this requirement can be very prohibitive. To adjust the encryption properties for each server, use the XenApp Connection Configuration administration tool, as defined later within this book. The advantage to this approach is that all ICA connections communicating with this configuration are encrypted, whether through any published application or a custom ICA connection.

The next option involves setting encryption options per published application. The primary advantage of using this method is that it applies to all servers using the published application. Any user connecting through this application will use the specified encryption level across all servers in the XenApp farm. Another advantage is that you can specify different levels of encryption for each published application. For example, if a user connects to a financial application, you might require him to use 128-bit encryption. At the same time, other users connect to a word processing tool over slow network links. For these connections, you may opt to force users to use only 40-bit encryption. To configure published application encryption, you can select the encryption strength when the application is created.

If the settings were managed at the server level, as described in the last section, you would have to separate the user connections by server or configure each client independently to allow this to work. By configuring encryption for each published application, you can easily manage multiple encryption levels simultaneously, without the users knowing the difference. The third option involves specifying an encryption level for the ICA client device. By default, the ICA client attempts to use whatever encryption strength is requested by the server. You can configure the ICA client to use different encryption strengths if the server or published application allows it. For example, if the server connection encryption strength is set to 56-bit, the ICA client cannot connect unless it is using 56-bit or higher. If you're connecting as an administrator to your XenApp farm across the Internet, you might prefer to use 128-bit encryption to ensure that traffic is secured.

Using HTTPS

Encrypting the ICA client traffic secures the session information, but you must also consider accessing published applications via the Web Interface. By default, standard Web browsing with HTTP access from client devices accessing via the Web Interface transmit data in cleartext. If you want to ensure that your communications are completely secure, you must consider using SSL on the Web server hosting the Web Interface. SSL is an industry-standard encryption technology that is application independent and works well with Web-based solutions. By configuring your Web server to support SSL technology in the form of HTTPS, client to Web server communication will now be secure. Once a server certificate is installed, the Web site can digitally sign and encrypt packets as they are sent between the client device and the Web server.

ICA Encryption (Secure ICA)

Secure ICA is the oldest method for securing communications between XenApp Servers and clients. Back in the days of WinFrame, Secure ICA was considered an additional feature pack that would be purchased from Citrix and then licensed on your WinFrame servers to allow the administrator or users to secure the connection. Today, this functionality is built in to the product although not enabled by default. The current implementation of Secure ICA is simply referred to as Encryption in the XenApp suite. Secure ICA is the legacy name, but we will continue to use it here to allow us to better differentiate the concepts. The use of the legacy name will be helpful, as we will be discussing and comparing several types of encryption in this section.

NOTE

Secure ICA uses symmetric key algorithms based on the work by [OR of] Ron Rivest. The algorithms used today are RC5 (meaning either Ron's Code or Rivest Cipher, and this being the fifth derivation thereof). Symmetric algorithms are also known as "shared" key algorithms and are designed for speed of use. For more information, visit Ron's homepage at http://theory.lcs.mit.edu/~rivest/.

To implement Secure ICA, we must first understand how it works. Secure ICA is a feature of XenApp server and client that allows for complete encryption of all data flowing through ICA packets between the client and server. Traffic such as screen updates, mouse movements, keyboard input, and print jobs that are redirected to the client's printers can all be encrypted. Secure ICA does not encrypt other types of data (such as Web browsing, unless it occurs within an ICA session). Secure ICA supports 40-, 56-, and 128-bit encryption settings, plus an option for 128-bit at log-in time only and then drops to 40-bit for the rest of the session.

NOTE

The requirements governing the use of encryption change on a regular basis. Today, the United States government has "relaxed" the restrictions of the use of 128-bit encryption continuously from outside the United States and its territories. However, prior to implementing an encryption-based solution for remote access security, check with all governmental parties involved, as several nations prohibit the use of encryption (France). Moreover, it is illegal to "export" the encryption to certain countries, especially those considered enemies of the State (Libya, North Korea, etc.).

The impact of choosing to encrypt data on modern servers and clients is negligible. Provided the server and the client have enough horsepower to perform the encryption and decryption necessary, this can be a very easy solution to securing access to XenApp. Typically, only legacy hardware or low-end thin clients will notice a difference in performance between a Secure ICA versus standard ICA session. Secure ICA can be enabled at various locations in the XenApp toolset, at the network interface level, at the published application level, or via XenApp policies. Alternatively, clients can request that connections to servers or published applications be encrypted at a higher level, regardless of settings on the server.

NOTE

A user may ask for a session to be encrypted or ask for a higher level of encryption for an already encrypted session. However, if the server's connection or published applications require a minimum level, then the user will not be allowed to connect at a lower level of encryption.

Secure ICA can be implemented internally to your network or externally. It uses the same Transmission Control Protocol (TCP) port, 1494, as a non-encrypted ICA session. In many early solutions (prior to Secure Gateway or Access Gateway), many administrators would opt to use Secure ICA to encrypt their otherwise open solutions of network address translator (NAT)/PAT and proxy servers. Prior to Web Interface and XenApp server policy, some administrators chose to implement multihomed servers to allow for control of when and how to encrypt data.

In this scenario, sessions originating from the production network would not be required to use encryption. External (nontrusted) network users would be denied connections if they didn't connect using encryption. As previously mentioned, Secure ICA can be required on the network interface (via the XenApp Connection Configuration utility), on a published application by published application basis, or via XenApp policy. Published applications and XenApp policy are "automatic" in allowing clients to connect with no changes required on the client end. If we configure XenApp policies to require encryption, then any connection created (except those already published applications with encryption or policies requiring encryption) will require a setting adjustment on the client. Let us begin by looking at the client side. Remember, a client can request encryption at any point (even if it isn't required from the server side). For clients to request encryption, they will have to change the configuration of their application sets or their custom connections.

Using the SSL Relay Service

Another security issue to consider is the way in which traffic is passed between the Web Interface server and the XML service on XenApp. In a process that is similar to standard Web traffic, data is transmitted in cleartext. This becomes a security concern when traffic between the Web Interface server and the XenApp farm is insecure. For example, many organizations will place a Web Interface server in the DMZ, or demilitarized zone, while maintaining a XenApp server farm in a more secure network. In the event XenApp is not located in a secure network environment, the use of the SSL Relay service will help to mitigate the security concern for unencrypted traffic. Although the password is slightly encrypted, it does not provide a secure alternative to the encryption methods discussed in this chapter. To assist you with this problem, Citrix has developed the SSL Relay service. This service allows you to configure all traffic passing between Web Interface servers and a XenApp server to use SSL encryption.

Secure Gateway

Another methodology developed by Citrix allows the tunneling of all ICA client traffic using industry-standard security protocols such as Secure Sockets Layer (SSL) or Transport Layer Security (TLS). Citrix has developed a solution known as Secure Gateway that encrypts all XenApp client traffic such as ICA packets via industry-standard Internet encryption protocols to simplify the management of a secure infrastructure throughout your network. For example, by deploying a Secure Gateway in the corporate DMZ, the firewalls protecting your network from the Internet must only be configured to allow SSL packets to the Secure Gateway server from any ICA client. The Secure Gateway server will manage the connectivity and encryption across the public Internet and mask the XenApp farm. Not only does this provide a simplified security solution, it hides the server farm from potential intruders on the Internet. Although this offers security to the ICA clients, once the traffic passes through the Secure Gateway it is no longer encrypted. It is recommended to use one of the many other encryption techniques for the TCP\IP packets from the Secure Gateway to the XenApp farm.

The Secure Gateway is made up of the following components:

- **Secure Gateway Server** Central server that acts as "gateway" to the XenApp farm. The Secure Gateway Server acts as the middleman and validates the ticket provided by the STA.
- **Secure Ticket Authority (STA)** Creates a ticket for each session offering a more secure access methodology. With Citrix XenApp Server 4.5, the STA is handled by the XML service on the XenApp Server.
- **Citrix XML Service** Provides the interface between a XenApp server and the Web Interface server.
- **XenApp Farm** XenApp server farm provides published applications via the ICA Client.

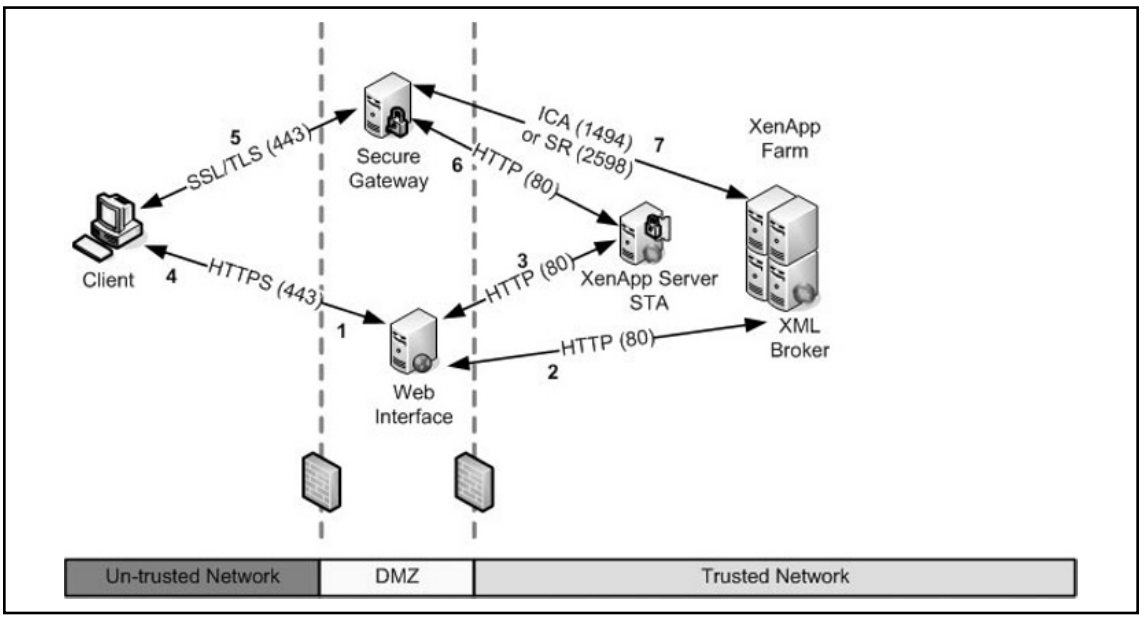
The typical Secure Gateway sequence, shown in Figure 4.32, follows:

1. A remote user launches a Web browser and connects to a Web Interface server on port 80 (HTTP) or port 443 (HTTPS). The Web Interface portal requires the user to authenticate using valid user credentials.
2. The Web Interface uses the user credentials to contact the Citrix XML Service, on port 80, running on a XenApp server and obtains a list of applications that the user is authorized to access. The Web Interface populates the Web portal page with the list of published applications

that the user is authorized to access. The communications so far are the normal sequence of events that occur when a Web Interface server is deployed to provide ICA client users with access to published applications.

3. When the user clicks on a link for a published application, the Web Interface sends the IP address for the requested XenApp server to the STA, also located on a XenApp server, and requests a Secure Gateway ticket for the user. The STA saves the IP address and issues the requested Secure Gateway ticket to the Web Interface.
4. The Web Interface generates an ICA file containing the ticket issued by the STA, and then sends it to the client browser. Note that the ICA file generated by Web Interface contains only the IP address of the Secure Gateway server. The address of the XenApp server(s) that the ICA client eventually connects to is not exposed.
5. The browser passes the ICA file to the ICA client, which launches an SSL connection to the Secure Gateway server. Initial SSL handshaking is performed to establish the identity of the Secure Gateway server.
6. The Secure Gateway server accepts the ticket from the ICA client and uses information contained in the Secure Gateway ticket to identify and contact the STA for ticket validation. If the STA is able to validate the ticket, it returns the IP address of the XenApp server on which the requested application resides. If the ticket is invalid or has expired, the STA informs the Secure Gateway server, and an error message is displayed on the ICA client device.
7. On receipt of the IP address for the XenApp server, the Secure Gateway server establishes an ICA connection to the XenApp server. After the ICA connection is established, the Secure Gateway server monitors ICA data flowing through the connection, and encrypts and decrypts client-server communications.

Figure 4.32 How the Secure Gateway Works

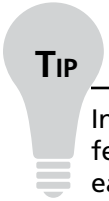


Clients that Can Support Encryption

To use encryption technologies, the ICA client software must be able to negotiate encrypted sessions. To accomplish this task, you must run a minimum version of 6.01 of the Citrix ICA client software. Additional enhancements continue to be released and it is recommended to use the Citrix ICA client version 10.x or higher. Using the client upgrade database can relieve the administrative overhead of managing client versions because the database can be used to deploy the version you want to use. For more information about the client upgrade database, download the Citrix Administration Guide available at http://www.citrix.com/.

Explaining IMA Encryption

A new feature included with XenApp is IMA encryption. This feature utilized the AES encryption algorithm previously discussed in this chapter to protect sensitive data in the IMA data store. IMA encryption is a farm–wide setting that applies to all XenApp servers once it is enabled. Therefore, once you enable IMA encryption, you must ensure that it is enabled on every XenApp server in your farm.



TIP

In environments that have a requirement for increased security, the IMA encryption feature should be utilized to protect sensitive data in the IMA data store. It is much easier to enable IMA encryption during your initial installation of XenApp than after you already have your farm installed and configured.

To enable IMA encryption, you must generate a key which is then used by all servers in your XenApp farm. You can specify the key before or during your setup. IMA encryption consists of the following components:

- **CTXKEYTOOL** This command line program is also known as the IMA encryption utility. You use this tool to manage and generate key files.
- **Key File** The key file contains the encryption key used to encrypt sensitive IMA data.
- **Key** Using the CTXKEYTOOL, you load the key you created during setup that is saved in the key file.

When using IMA encryption, Citrix recommends that you keep a backup of the farm key in a safe and secure location.

Summary

Maintaining the security of any system is not an easy task. Much thought and consideration must be given to the benefits of having a secure system weighed against the productive ability to use that system. When applying security measures to your system you must keep in mind that the system exists for one purpose and that it is to be used by clients that need to be productive. If you fail to take into account the parts of the CIA triad outlining the confidentiality, integrity, and availability of a system, then you will most certainly be looking for a new job. You can make your system so secure that not even you can use it. The XenApp Security Model is provided as a tool to assist you in completing the task of securing your system while at the same time maintaining the availability and performance level required by your organization. The model breaks down XenApp security into individual components, each with its own set of resources which you can use to make your XenApp environment more secure.

As an administrator it is your job to know and understand your network, to know which authentication mechanisms are the most appropriate for your organization, and to know which deployments will provide the best security while providing optimal availability. Make use of the resources that are readily available to you that can make your job easier. Microsoft has numerous tools available that can be used in assisting you to harden the underlying Windows 2003 server operating system on which XenApp is installed. Perhaps the best tool (probably the most dangerous in the hands of the inexperienced) is the Security Configuration Wizard that you can use to quickly and easily disable unnecessary services, remove unwanted IIS virtual folders, block unused ports, configure audit settings, and lock down access to critical system files. Other resources should be fully investigated and researched to see if they will be a good fit in your environment.

Understanding the IT methodology presented in ITIL can help you establish a firm security posture by making use of the processes and procedures it outlines. The ITIL concept is only briefly discussed in this book but contains volumes of information that is continually updated. It would be in your best interest to learn more about this methodology. Implementation of third-party products that support the ITIL approach by providing services such as automated patch management, compliance measurement, configuration management, compliance assurance, and remediation will help to make your job easier as a XenApp administrator.

Solutions Fast Track

Defining the XenApp Security Model

- ☑ The Citrix XenApp model consists of the following six layers: XenApp Servers (implementing computer group policy objects, locking down the base operating system), Published Applications (implementing application security, using Windows security groups, implementing NTFS permissions on application executables), ICA Connection (security on the ICA connection, encryption), Network Configuration (using IPSec, router ACLs, enforcing security boundaries), Client Devices (using only approved client devices to connect, using only approved client software to connect), End Users (establishing a "need to know" user access, no unauthorized "admin" type access, implementing user group policy objects).

- ☑ Defining farm and network security boundaries can assist you in using many of the security tools provided by Microsoft and other vendors to help you successfully secure your XenApp server.
- ☑ You should adopt and follow security guidance provided by trusted and established resources such as Microsoft, the National Security Agency, the SANS Institute, and others. In addition, adopting sound information technology methodologies such as ITIL will greatly improve your overall security posture and organizational productivity.

Understanding Types of Deployments

- ☑ Using the XenApp Security Model as a guide, you can determine the best type of deployment for your organization and best approach to secure that deployment. Insuring that your deployment follows the principles of the CIA triad of confidentiality, integrity, and availability will provide you with the proper balance of security in relation to productivity.
- ☑ There are several types of XenApp deployments that you can implement. The most common is Internal Deployment using SSL Relay; External Deployment using only a Web Interface server and a Secure Gateway server; External Deployment using a Web Interface server, Secure Gateway proxy, and a Secure Gateway server; External Deployment using SSL Relay with a Web Interface server; and deployments using a combination of all methods.
- ☑ Employing the use of IP Security policies can provide greater security to your deployments by restricting access to defined ports.

Authentication Methods

- ☑ Kerberos is supported in the XenApp environment, but there are specific requirements that must be met before it will work successfully. Requirements for Kerberos Support with XenApp are the following: XenApp on Windows 2003, ICA Client Version 8.x or higher, client\server connections must be in same domain or have trusts between domains, servers must be "trusted for delegation," SSPI requires the XML Service, DNS service address resolution to be enabled for the server farm, or reverse DNS resolution to be enabled for an AD domain. Microsoft best practices recommend not using Kerberos for authentication via the Internet.
- ☑ An authentication factor is a piece of information and process used to authenticate a person's identity for security purposes. Two-factor authentication (2FA) is an authentication mechanism based on two pieces of information: something you have, such as a smart card or token id, and something you know, such as a Personal Identification Number (PIN). XenApp fully supports the implementation of 2FA authentication mechanisms.
- ☑ You can optimize your XenApp environment by utilizing multihomed servers to segregate server functions, such as dedicating a single NIC for private traffic for server-to-server communication between XenApp resources and domain controllers, and by having another network card configured for communicating with DMZ components such as the Web Interface or the Secure Gateway.

Encrypting XenApp Server

- ☑ Citrix XenApp provides built-in capability to help secure server and client communications from intruders. By implementing the use of various encryption technologies available to XenApp, you can provide secure server-to-server and secure client/server communication that can be protected against intruders.
- ☑ Citrix XenApp offers multiple encryption techniques providing flexibility in managing secure sessions. Encryption is a key component ensuring secure communications between the XenApp farm and the ICA client. There are several ways to configure encryption: encrypting traffic at the server level, setting encryption for each published application, and setting encryption on an individual client basis. Citrix XenApp fully supports encryption standards such as FIPS140-2 and AES.
- ☑ A new feature included with XenApp is IMA encryption that utilizes AES encryption to protect sensitive data in the IMA data store.

Frequently Asked Questions

Q: Why is XenApp security broken down by the XenApp Security Model?

A: Implementing security based on the XenApp Security Model will assist you in protecting your network from different threats and different users.

Q: What are the components of the XenApp Security Model?

A: Servers, Published Applications, ICA connections, network configuration, client devices, end users.

Q: How can creating a Farm Boundary or network diagram assist you?

A: A "picture is worth a thousand words" and by having a quick glance view of your network boundaries and assets, you can quickly ascertain your "weak" points of security.

Q: What are some resources available to assist in server security hardening?

A: Microsoft tools such as Security Configuration and Analysis Tool, Baseline Security Analyzer, Security Assessment Tool, IIS Lockdown Tool, Security Configuration Wizard, Security Templates, Group Policy Objects.

Q: What is ITIL?

A: ITIL stands for Information Technology Infrastructure Library which is a methodology for IT management that covers areas such as configuration management, change management and security configuration and remediation.

Q: What are the different types of XenApp Server deployments?

A: Internal with SSL Relay, External (single-hop), External (double-hop), External with SSL Relay, and Combination Deployment

Q: What is the CIA triad?

A: The CIA triad stands for confidentiality, integrity, and availability.

Q: Which deployment(s) can provide end-to-end 128-bit encryption?

A: Internal SSL Relay, External deployments using Secure Gateway, or a Citrix Access Gateway used in conjunction with SSL Relay

Q: Are the Secure Gateway or SSL Relay features configured by default installation?

A: No. SSL Relay, though installed on the XenApp server, is not configured by default. Secure Gateway is a separate product that must be installed and configured separately from XenApp server.

Q: Can smart card authentication be utilized in a double-hop deployment?

A: No. Citrix does not support smart card authentication for this type of deployment or for a single-hop deployment where the Secure Gateway is placed in front of the Web Interface.

Q: Does XenApp support the use of Smart Cards?

A: Yes. Smart Card authentication is supported in most types of XenApp deployments.

Q: Should you use Kerberos authentication for any XenApp component that will be connected to the Internet?

A: No. Microsoft best practices recommend that for computers connected to the Internet, do not use Kerberos as an authentication method.

Q: What is the definition of two-factor authentication?

A: Two-factor authentication consists of something you have (like a smart card) and something you know (like a PIN).

Q: Does XenApp support the use of biometric devices to provide multifactor authentication?

A: Yes, but to enable authentication mechanisms utilizing biometric devices requires the installation of third-party software.

Q: What is the benefit of AES encryption?

A: In addition to the increased security that comes with larger key sizes, AES can encrypt data much faster than Triple-DES.

Q: Which XenApp components are FIPS-140 compliant?

A: Citrix Clients for 32-bit Windows (including Program Neighborhood, Program Neighborhood Agent, and the Web Client), Secure Gateway, XenApp, Citrix SSL Relay, Web Interface, Citrix Access Gateway, and Citrix NetScaler.

Q: If a user creates a custom ICA connection and configures the session to be encrypted at a lower level than what XenApp policies have already applied, will the user be able to initiate the session?

A: No. If a XenApp policy has defined a minimum encryption level, then the user will not be allowed to connect at a lower level of encryption.

Q: What is the difference between symmetric and asymmetric encryption?

A: Symmetric encryption requires that each individual or device that accesses encrypted data possess a copy of a key, commonly referred to as shared-key encryption. Asymmetric encryption uses two keys to encrypt data and is known as *public key encryption*.

Q: What is the command line tool used for creating and managing keys enabling IMA encryption?

A: The command line tool is CTXKEYTOOL.