Global Information Assurance Certification Paper

Internet Protocol Security (IPSec) in Windows 2000: An Introduction to Securing Datagrams and TCP/IP Traffic with Pre-Defined Policies and Simple Filtering Rules

SANS/GIAC Level 2 Windows Security Practical Option 1
New Orleans 2001

Richard Genova
9 May 2001

Table of Contents **Page**

1. Introduction

Windows NT4 allows only rudimentary TCP/IP filtering functionality and no native encryption functionality for files or data at any level of the OSI seven-layer TCP/IP stack. As part of the overhaul of its flagship network operating system, Microsoft has addressed these issues with some strong security enhancements in Windows 2000. Thus Windows 2000 offers the Encrypting File System for local files on NTFS volumes, and it also provides for the encryption of IP packets at the network level through its implementation of Internet Protocol Security (IPSec) to secure data against unauthorized remote users and hacker sniffing attacks. IPSec is the focus of this paper which is intended as an introduction to applying the technology in Windows 2000. Microsoft describes its IPSec solution as providing "end-to-end security for network communications—in the form of confidentiality, integrity, and authentication—using public-key technology to protect individual IP packets." This means secure links for private network users within the same Windows 2000 domain or across any trusted domain in the enterprise.

The Microsoft implementation of IPSec is based on the Internet Engineering Task Force (IETF) RFC 2401, "Security Architecture for the Internet Protocol." There are various scenarios for using IPSec in Windows 2000 that allow users' application data to be transparently exchanged by providing an authenticated, secure channel using one of three protocols: Internet Key Exchange (IKE, see RFC 2409), Authentication Header (AH, see RFC 2402) and Encapsulating Security Payload (ESP, see RFC 2406). Windows 2000 supports integration of IPSec with the Active Directory service to deliver central control of policy-based security administration. IPSec policies can also be stored locally in the registry for stand-alone computers and servers which are not part of a trusted Windows 2000 domain. Microsoft and Cisco Systems jointly developed IPSec and related services in Windows 2000 with the stated goal of providing simplified deployment and manageability of privacy, integrity and authenticity for network traffic by addressing threats from spoofing and tampering of IP packets and denial of service (DOS) attacks.

Microsoft's use of IPSec in securing Windows 2000 data has been praised widely within the IT industry. For example, Jeff Schmidt states in his Microsoft Windows 2000 Security Handbook that IPSEC "provides both the strength and flexibility to protect virtually any type of data communications imaginable. That includes communications between internal computers, remote sites, extranets, and dial-up clients" (pg. 182). Similarly, Paul Robichaux in his guest Technet column "Robichaux on Security – April, 2000" states that IPSec is "a very flexible and capable security protocol" which "delivers some important capabilities, and it does so in a way that makes it easy for you to deploy and manage it."

Praise from these industry experts for IPSec in Windows 2000 was partly earned through the OpenHack and Windows 2000 test attack sites which allowed hackers to try their skills in compromising a Windows 2000 system. Prerelease versions of Windows 2000 were hosted at http://www.windows2000test.com and http://www.openhack.com in late 1999 and the public was invited to try hacking them. The sites were not compromised. Of course, IPSec alone didn't deserve all the credit for successfully resisting attacks, but it was an important part of a total security package. Some hackers thought mistakenly that they were confronting a firewall when their port scans failed because their packets were dropped. In reality it was IPSec filtering rules

implemented at the attacked host sites that overcame the lack of security features in the TCP/IP protocol and safeguarded the sites.

2. IPSec Overview

Before diving into using the IPSec security profiles in Windows 2000, it's important to see what respected security and cryptography experts who have examined and worked with IPSec have stated about its functionality and limitations so as to understand better what Microsoft has accomplished with its implementation of IPSec. Perhaps the severest critics of the whole RFC-based standard are Niels Ferguson and Bruce Schneier of Counterpane Internet Security, Inc. They described IPSec as "a great disappointment to us," and added that "given the quality of the people that worked on it and the time that was spent on it, we expected a much better result. We are not alone in this opinion; from various discussions with the people involved, we learned that virtually nobody is satisfied with the process or its result."

Although Ferguson and Schneier were focused on the cryptographic properties of IPSec rather than on the integration aspects specific to Windows 2000 or any other operating system, the source of Ferguson and Schneier's disappointment is "too many options and too much flexibility" in IPSec. In contrast, Jon Hollandsworth wrote in his "Overview of IPSec Manageability and Security" (http://www.sans.org/infosecFAQ/encryption/IPSEC.htm) that "the beauty of IPSec lies in its extensibility to new and stronger encryption and authentication methods." Ferguson and Schneier put the emphasis of their criticism on this very same underlying complexity. "There are often several ways of doing the same or similar things…. As we all know, this additional complexity and bloat is seriously detrimental to a normal (functional) standard. However, it has a devastating effect on security standards."

A common misunderstanding with IPSec which Ferguson and Schneier argued against in their paper is that "IPSec provides IP-level security, and is thus essentially a VPN protocol. Yet we hear about people trying to use it for application-level security, such as authenticating the user when she tries to get her email. IPSec authenticates packets as originating from someone who knows a particular key, yet many seem to think it authenticates the original IP address as that is what they can filter on in their firewall. Such misuse of IPSec can only lead to problems." Their criticisms didn't end there. Like many others, Ferguson and Schneier bemoaned the dense and tangled RFC documentation: "Various parts of the IPSec documentation are very hard to read. For example, the ISAKMP [Internet Security Association Key Management Protocol] specifications contain numerous errors, essential explanations are missing, and the document contradicts itself in various places." The danger is interoperability problems that compromise the spread of reliable technical information for properly implementing IPSec. This is a major issue for already overworked network administrators who are often security administrators too. These administrators need a straightforward installation of a security protocol or policy to deploy for a given situation without becoming cryptographic experts too. Here Microsoft has greatly aided us, by providing default IPSec policies to use which will be demonstrated later in this paper.

Despite the criticisms leveled at the standard, IPSec as implemented in Windows 2000 offers network administrators and security officers another useful and important tool in the never-ending battle to provide integrity and confidentiality for application data that runs over IP, much as Secure Sockets Layer (SSL) can provide security for an application that uses a connection-

oriented transport. Both protocols use a handshake to negotiate keys and parameters and then a data transfer is allowed. One advantage of IPSec, however, is that unlike Secure Sockets Layer (SSL) which requires replacing Windows socket calls with SSL socket calls, IPSec can be added without changing applications.

Another tool sometimes compared to IPSec is the Secure Shell (SSH). Among Unix, Linux and OpenBSD users, SSH is rapidly replacing Remote Shell and Telnet. Commercial and open source ports of SSH are also being offered for use on Windows networks. SSH takes data that is sent by a computer to the network and automatically encrypts it. When the data reaches the recipient, SSH automatically decrypts it. SSH uses a client/server architecture and provides the user with a transparent experience. One advantage that is claimed for SSH over IPSec is that it is a simple application program and thus is easier and quicker to deploy. However, IPSec authentication and encryption is deployed at the layer 3 Network (IP) level which is lower in the network stack than what SSH addresses. Thus IPSec gives more basic protection and it overcomes SSH's limitations, such as its inability to forward dynamic ports. Moreover, at this time most ports of SSH to Windows don't implement the secure file copy feature available with the Unix variants.

IPSec in Windows 2000 will not replace the use of third-party hardware and software tools such as firewalls (i.e., Raptor, Checkpoint FW-1, etc.) and policy servers (i.e., Siteminder, SecureWay Policy Director, etc.) to achieve strong, layered perimeter security for bastion and DMZ hosts, especially those providing web-based application and services in large enterprise environments. IPSec in Windows 2000 gives us IP filtering rules that are more sophisticated than those provided by NT4. The Windows 2000 IP filters allow specifying ranges of addresses in the filter rules and more than one IP filter can be part of a security policy. However, firewalls and routers which perform dynamic filtering and stateful inspection of packets are superior to anything IPSec can offer. Nor does IPSec by itself provide the full range of features necessary in a total network and application security package that allows single sign-on access and which provides network and security administrators with the ability to control authentication, authorization and auditing across all users and applications on a corporate web site and intranet. When IPSec is used in combination with strong user access control, perimeter firewalls and physical security, it can contribute greatly to a strong, layered defense of your data's integrity and confidentiality.

There are caveats with the standard itself as Ferguson and Schneier's paper warns, but IPSec can and does mitigate spoofing and tampering of IP datagrams by authenticating packets, encrypting packets and using packet filtering to stop some DOS attempts at the IP level. Moreover, ***Windows 2000 IPSec can be used within the perimeter to improve data authentication and to add packet encryption between an application server and a host.*** A secure communication channel can be constructed to connect a department or workgroup that handles highly sensitive data within the enterprise to the intranet and to shield the data with encapsulation and encryption. The department or workgroup's resources will not be visible to users outside the protected groups who don't have proper access permission and traffic will be over the LAN rather than a dial-up in this special type of secured channel.

I'm calling this a "special type of secured channel" rather than an "internal VPN" as some writers have, because I want to highlight the ability of Windows 2000's implementation of IPSec

for use inside the enterprise firewall to secure LAN traffic. This is a significant capability, because as security administrators are constantly trying to underscore for senior management, most of the "hacking" and compromising of data occurs inside the firewall despite the typical expenditure of the lion's share of enterprise security funds and efforts for perimeter defense. The second reason I make this distinction is because I prefer to use Microsoft's narrow definition of a VPN which states that a "VPN refer(s) to providing security across a public or untrusted network infrastructure. This includes secure remote access from client-to-gateway, either through internet connections or within private or outsourced networks [and] secure gateway-to-gateway connections, across the internet or across private or outsourced networks."

As Windows 2000 market penetration of corporate systems continues to advance, Microsoft's IPSec features will become more attractive beyond purely VPN solutions (i.e., Layer 2 Tunneling Protocol as implemented in Windows 2000 for remote client to gateway access over the Internet defaults to IPSec for encryption purposes). Microsoft has supplied the network administrator with predefined security policies and extensive documentation to ease the burden in spreading IPSec encryption and filtering. Even tough-minded critics like Ferguson and Schneier conceded that "with all the serious criticisms that we have on IPsec, it is probably the best IP security protocol available at the moment. We have looked at other, functionally similar, protocols in the past (including PPTP) in much the same manner as we have looked at IPsec. None of these protocols come anywhere near their target, but the others manage to miss the mark by a wider margin than IPsec…. [which] is the current `best practice,' no matter how badly that reflects on our ability to create a good security standard."

3. Basic IPSec Operation: Protocols and Modes

One aspect of IPSec that should be understood before beginning deployment and which Ferguson and Schneier targeted as problematic, are the two modes of operation: transport and tunnel. Before two hosts can establish a secure connection, they must negotiate an encryption method to use after they've authenticated each other. This is accomplished with one or more Security Associations (SA). Ferguson and Schneier object that along with the two transport modes there is the additional problematic complexity of two different protocols: Authentication Header (AH) and Encapsulating Security Payload (ESP) so that two machines that wish to authenticate a packet can use a total of four different modes: transport/AH, tunnel/AH, transport/ESP with encryption and tunnel/ESP with encryption. They argue to eliminate transport mode and note that the ESP protocol allows the payload to be encrypted without being authenticated. "In virtually all cases, encryption without authentication is not useful…. We [Ferguson and Schneier] recommend that ESP authentication always be used, and only encryption be made optional."

For Windows 2000 IPSec, the SA is a connection identifier that is unique. The Internet Key Exchange (IKE) protocol is used by Windows 2000 to establish the SA needed for a typical bidirectional communication between two hosts. During the end-to-end communication between two IPSec-enabled computers, the IKE (*originally this was the Internet Security Agreement/Key Management Protocol and Oakley protocol in the early RFCs, i.e. ISAKMP/Oakley*) provides a way to build the security association. Following the key negotiation, the communicating computers negotiate the actual IPSec settings to use for the connection.

When the IPSec SA destination address is the final destination of the datagram, *Transport Mode* is used. If the SA is between two gateways--such as through firewalled segments of a network--*Tunnel Mode* is used. AH packets are only supported by Windows 2000 clients in a Microsoft networking environment. AH's mutual authentication capabilities guard against "replay" attacks. The AH contains a checksum on the entire datagram that is inserted after the original IP header in the IPSec datagram, thus providing data integrity and protection against replay attacks. In transport mode, the SA destination address is the final destination of a packet. This can secure client-server communications that contain sensitive data. Tunnel mode allows two security gateways such as two firewalls to tunnel IP in IP. It's now possible for end-to-end AH transport mode over the ESP tunnel mode. VPNs can be established to allow a remote client to establish a connection over the Internet or public network. (Inside the corporate network, the communications are no longer secured by IPSec from the security gateway to the target host.)

As Mel and Baker succinctly explain in Cryptography Decrypted (Addison-Wesley, 2001): "The current IPSec standard can be visualized as having two parts. The first part, IKE, manages authentication and key exchange. The second part manages the bulk encryption process. IKE is a two-phase protocol. The first phase sets up a secure authenticated communication channel; phase 1 establishes encryption parameters that are used to protect the second phase. The second phase makes encryption parameters that are used in IPSec part 2, bulk encryption. The result of this two-phase protocol key management is that it enables quick changes to encryption parameters." The result is that bulk encryption and message integrity occurs in four potential configurations:

ESP + Transport
AH + Transport
ESP + Tunnel
AH + Tunnel

As we shall see, Microsoft didn't eliminate the use of transport mode as Ferguson and Schneier urged, but they have simplified the use of IPSec if an administrator utilizes the pre-defined policies and the best practices recommendations provided by Redmond's experts. Moreover, IPSec can use both AH and ESP to ensure data integrity by creating multiple SAs, and you can customize the pre-defined policies or create your own from scratch. If you agree with Ferguson and Schneier, this complexity isn't always good, but if you appreciate Microsoft's approach to extending the RFCs governing IPSec for its use by Windows 2000 then you will appreciate that IPSec can provide multiple layers of security.

4. Encryption and HMACs

In the transport mode, only the data payload and the application and transport level protocol fields are encrypted, whereas in tunnel mode the entire datagram is encrypted. The actual encryption standards used are 56-bit Data Encryption Standard in Cipher Block Chaining mode (DES-CDC) and 168-bit Triple DES (3DES), which provide secret key algorithms for confidentiality. A random number is generated and used with the secret key to encrypt the data. Cipher Block Chaining hides patterns of identical data blocks within the packeting without

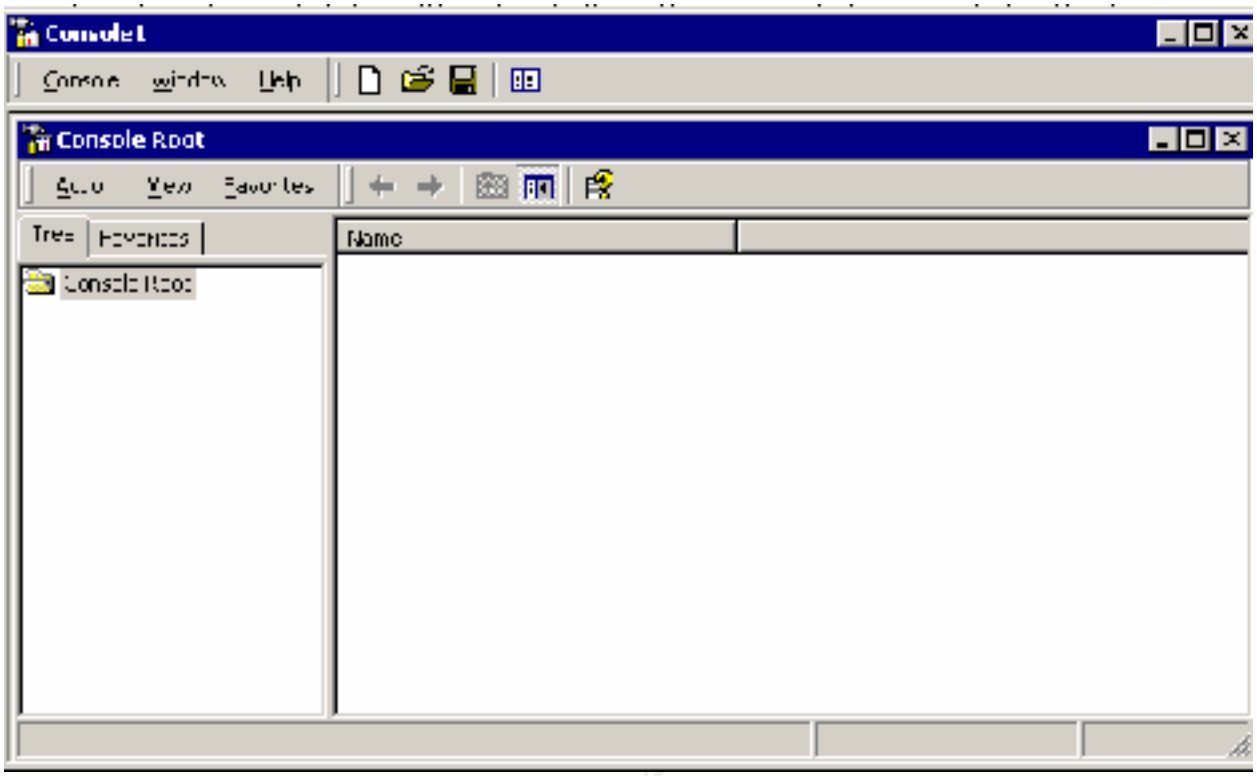
increasing the data size after encryption. Unless legal restrictions such as export rules prevent you from using 3DES, it is recommended to use the superior encryption of 3DES rather than plain DES because of the well-documented vulnerabilities of 56 bit DES to brute force attacks. Take ESP packets, for example. Before the ESP packets are sent, the data is encrypted by DES in Cipher Block Chaining mode or by 3DES, then the recipient computer with the shared, secret key becomes the only one capable of decrypting the transmitted data and modifying it. An attacker armed with a sniffer who manages to intercept the communication will not be able to decipher it.

The AH secures the packet integrity and authenticity. It does this by adding a header to each packet which contains a keyed hash or more technically speaking, a Hash Message Authentication Code (HMAC). The HMAC uses an algorithm that combines a secret key with a hash function. The hash function can either be the 128-bit Message Digest Function 5 (MD5) or the stronger 160-bit Secure Hash Algorithm (SHA-1). The HMAC can determine whether someone has tampered with a packet. When an application receives the data packet, the application rehashes the data and also hashes the secret key that the application knows to rederive from a Message Authentication Code (MAC). If the two MACs match, then the data hasn't been tampered with and it must be from another party who has access to the same secret key used to send the data. In effect, the HMAC provides a digital signature for the packet that is verified by the receiver.

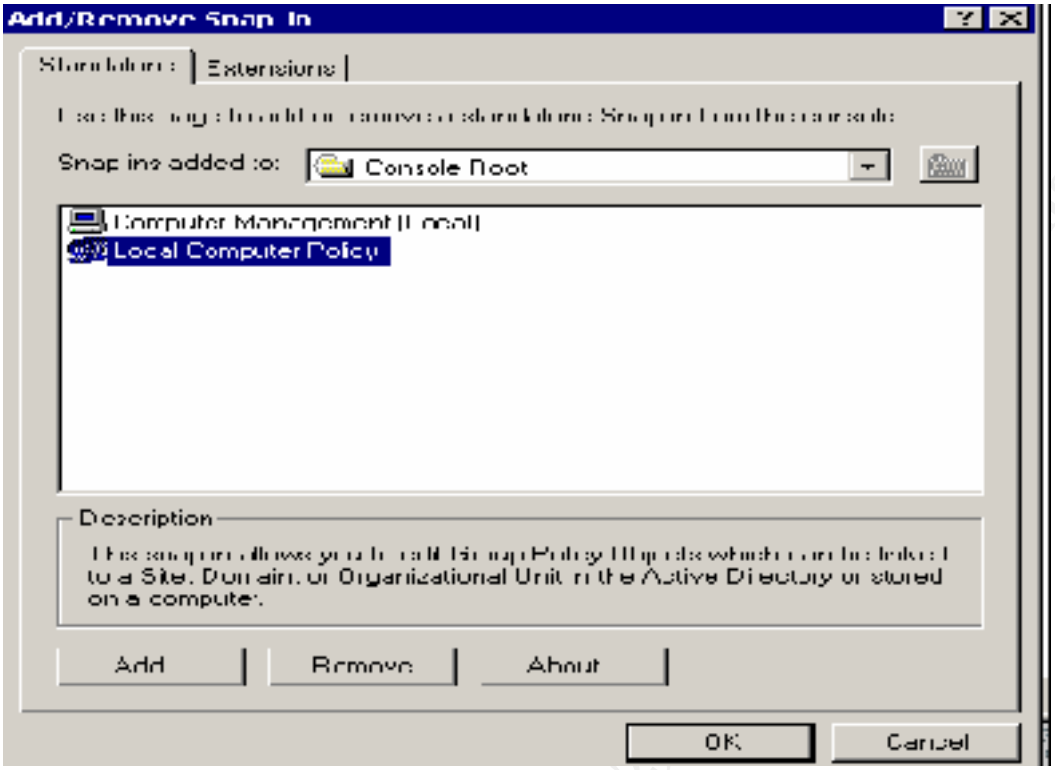
5. Deploying an IPSec Policy

Always keep in mind that IPSec is applied to machines, not users. It's one piece of a security solution in an enterprise. IPSec can be used to harden perimeter servers with another layer of security as long as the limitations of how it works are properly understood and the additional computational strain the encryption process introduces is factored. As mentioned, other possibilities for the intrepid network administrator of a Windows 2000 domain without downlevel Windows clients include using IPSec to secure data transmissions between departments or groups which need secure bi-directional data transfers between machine hosts within the LAN. As we will see in the walkthroughs, the easiest and fastest deployments of IPSec in Windows 2000 always utilize Kerberos authentication.

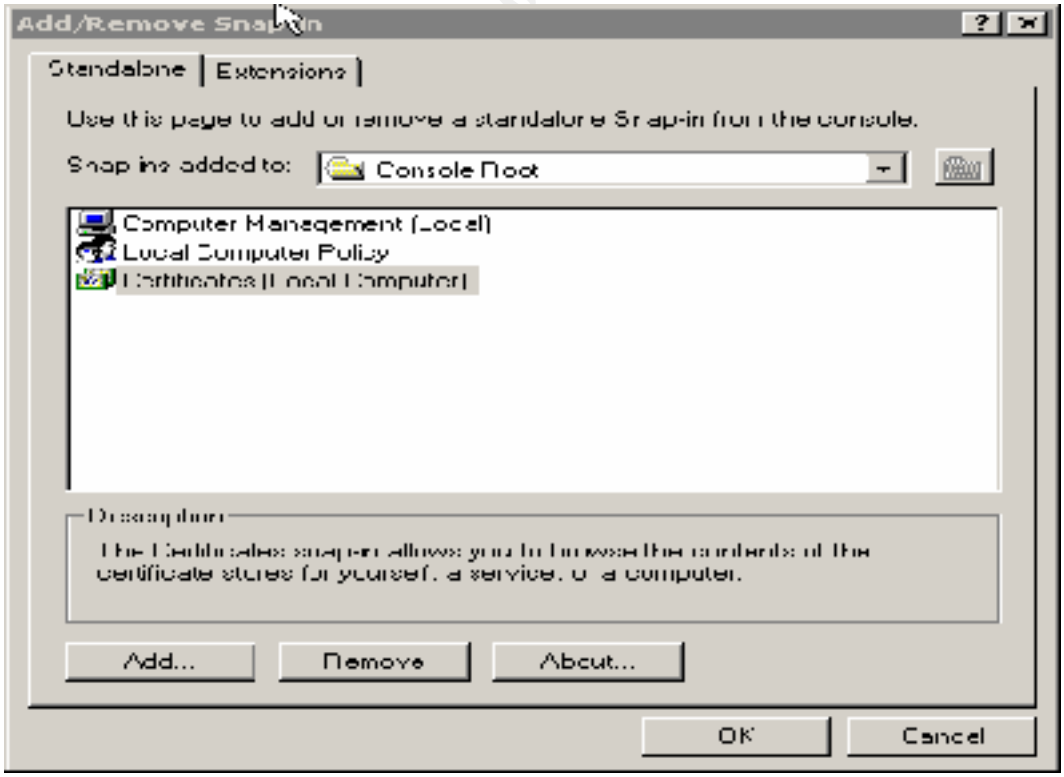
For the demonstrations provided in this paper there are two computers involved. **Server01** is a domain controller (Windows 2000 Advanced Server with Service Pack 1; its IP address is 192.168.1.99) and **Station** (Windows 2000 Server with Service Pack 1; its IP address is 192.168.1.33) is a member server in the **rgenova.net** Active Directory namespace. Before we can actually enable a pre-defined IPSec policy, we will create a **Microsoft Management Console (MMC) snap-in**. Begin in the run command text box by clicking **Start | Run**. Type MMC in the command text box.



Click the Console menu and select **Add/Remove Snap-in**. In the Add Standalone Snap-in dialog box, select **Computer Management**. Confirm this choice by clicking on Add. Confirm that **Local Computer** is selected, and select Finish. Go to the **Add Standalone Snap-in** dialog box, and select Group Policy, and click Add. Confirm that **Local Computer** is selected in the **Group Policy Object** box, click Finish. Verify that Local Computer is selected, and click on Finish. Close the **Add Standalone Snap-in** dialog box and then close **Add/Remove Snap-in** box.

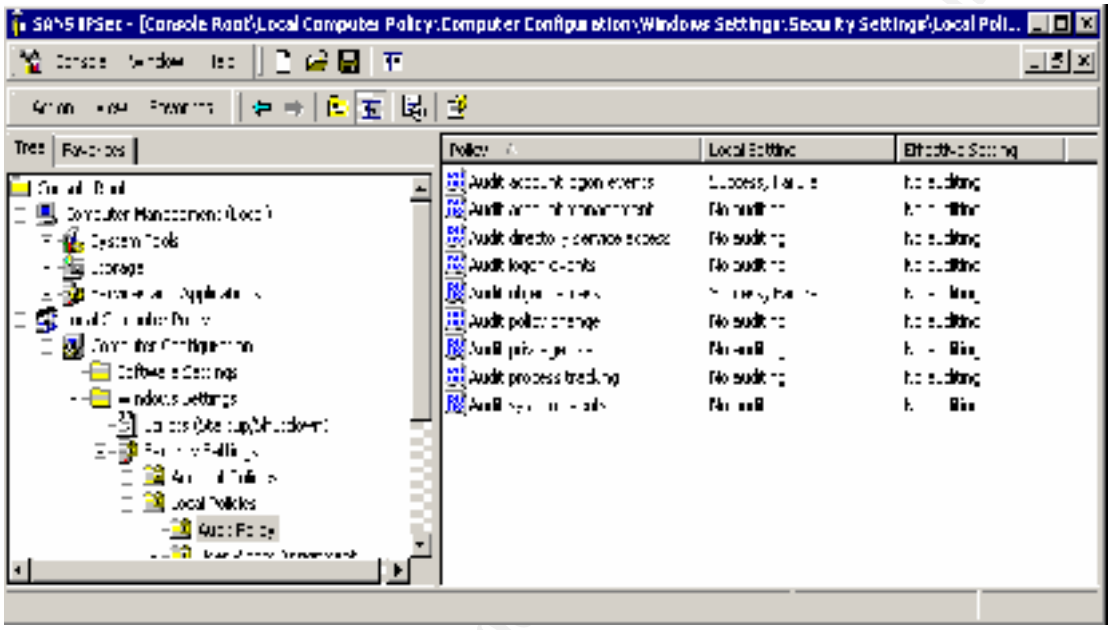


For a custom policy and interoperation among non-Windows 2000 hosts, we would repeat the above the sequence of steps to add **Certificates** because non-Windows 2000 hosts may not support Kerberos authentication.



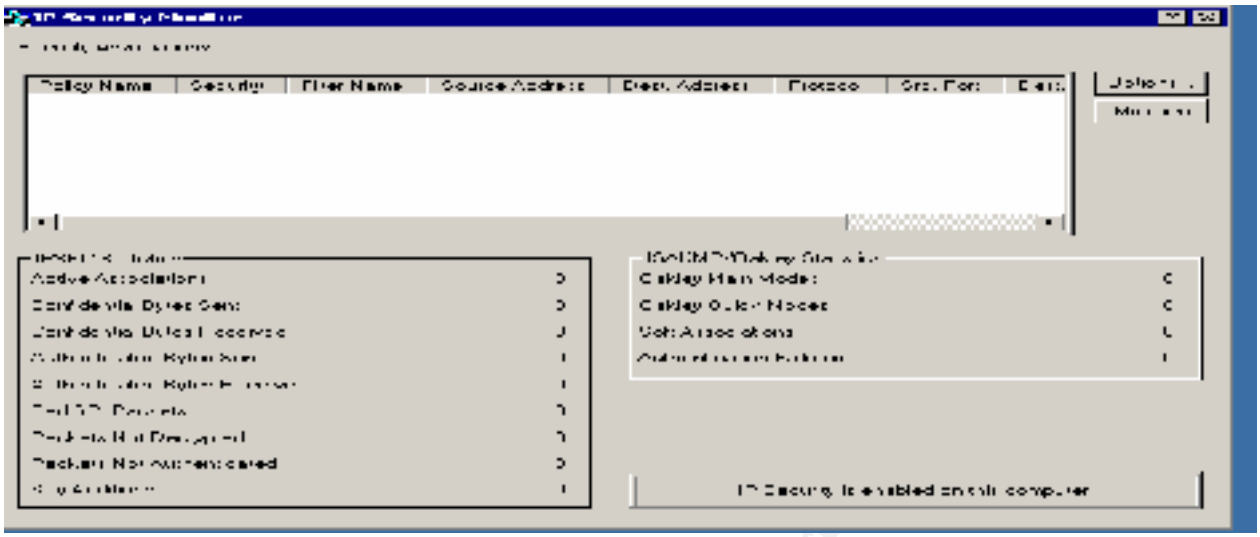
To enable Auditing:

Within the MMC console, launch **Local Computer Policy** and expand the tree. Go to **Computer Configuration**, **Windows Settings**, then advance to **Local Policies** and select **Audit Policy**. Double-click **Audit Logon Events** from the list of Attributes. Within the **Audit Logon Events** dialog box select Audit these attempts: **Success** and **Failed** boxes, click Ok. Select **Success** and **Failed** for **Audit Object Access**.



Invoking the IP Security Monitor

Next we want to use an important tool for verifying and troubleshooting our IPSec policy by configuring the IP Security Monitor. To start the configuration of the IP Security Monitor tool, launch Ipsecmon from the **Start | Run** text box by typing "**ipsecmon,**" then click OK.



You'll see the above IP Security Monitor screen after clicking OK in the Run command text box. Click the IP Security Monitor **Options** button and change the default **Refresh** value from 15 seconds to 1 second. Select and **minimize** the monitor window.

6. Configuring an IPSec Policy

IPSec policies are controlled by the policy agent, which passes the defined policy to the IPSec driver and sets the procedures for all aspects of the protocol in determining how to secure data and what security methods to use. To define IPSec policy for computers, you must have appropriate administrator rights to Group Policy or be a Local Systems Administrator.

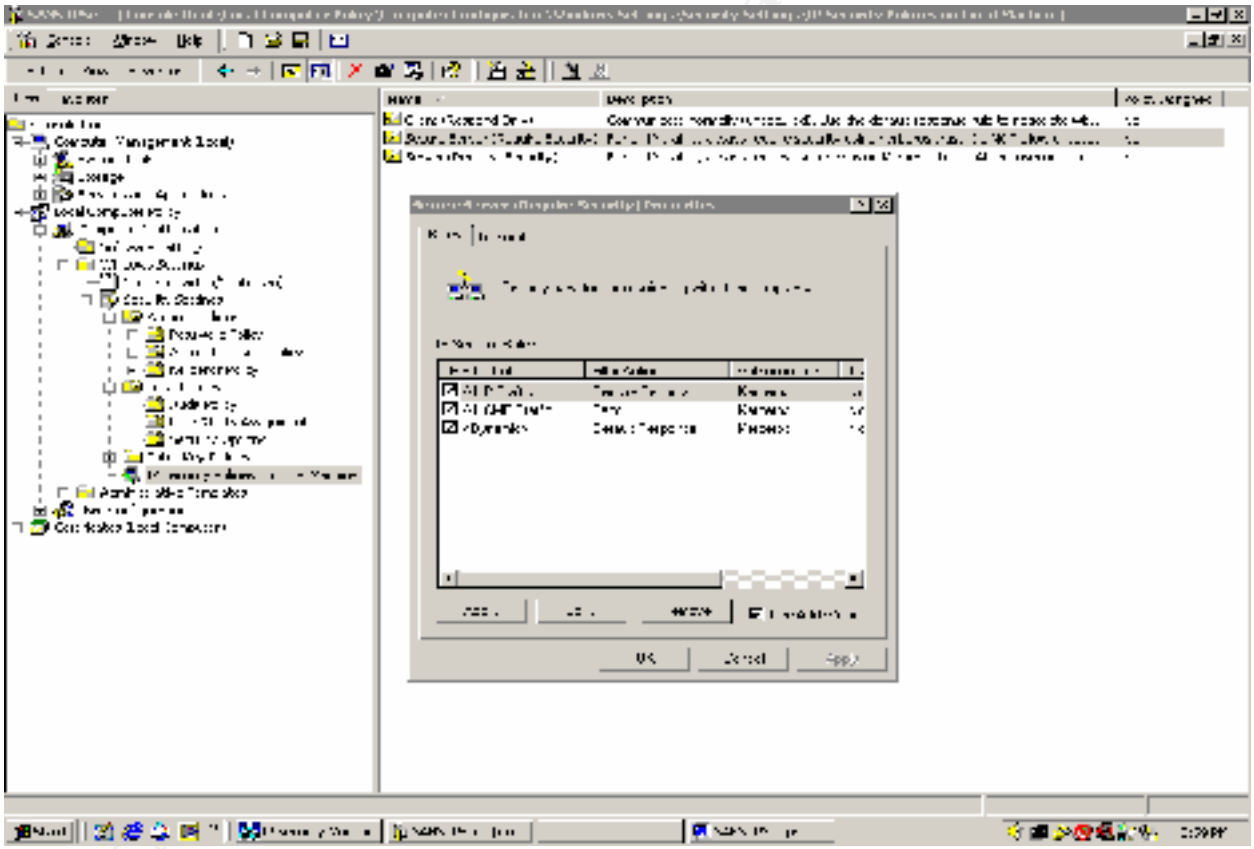
There are three main areas to configure for implementing IPSec in Windows 2000. They are IP Security Rules, IP Filter Lists, and IP Filter Actions. The IP Filter tells the IPSec driver what outbound and inbound traffic to secure based on IP address, port and transport protocol. The IP Filter List is the combination of one or more IP filters which cover a scope or range of network traffic. The Filter action is how the IPSec driver should secure the network traffic. The security method defines the security algorithms and types used for the authentication and key exchange. The IPSec Security Rules consist of the IP filters, filter actions, security methods, tunnel settings and connection types. Our IPSec policy can have multiple rules to protect each subset of network traffic individually.

Below, we are going to simplify our configuration chores for IPSec in Windows 2000 by using a Microsoft pre-defined IPSec policy. By default in Windows 2000 **there are three pre-defined IPSec policies: Client (Respond Only), Server (Request Security) and Secure Server (Require Security).** The Client policy is the least restrictive of the three and should only be used for computers that do not send secured data. This policy doesn't initiate secure communications. Only if the server requests security will the client respond by securing the requested protocol and port traffic with the requesting server. The Server (Request Security) policy can be used on a client or server that wants to initiate secured communications. It will attempt to protect all outbound network traffic and it will accept unsecured inbound data packets but will not resolve them until IPSec requests security from the sender for all subsequent transmissions. The strictest

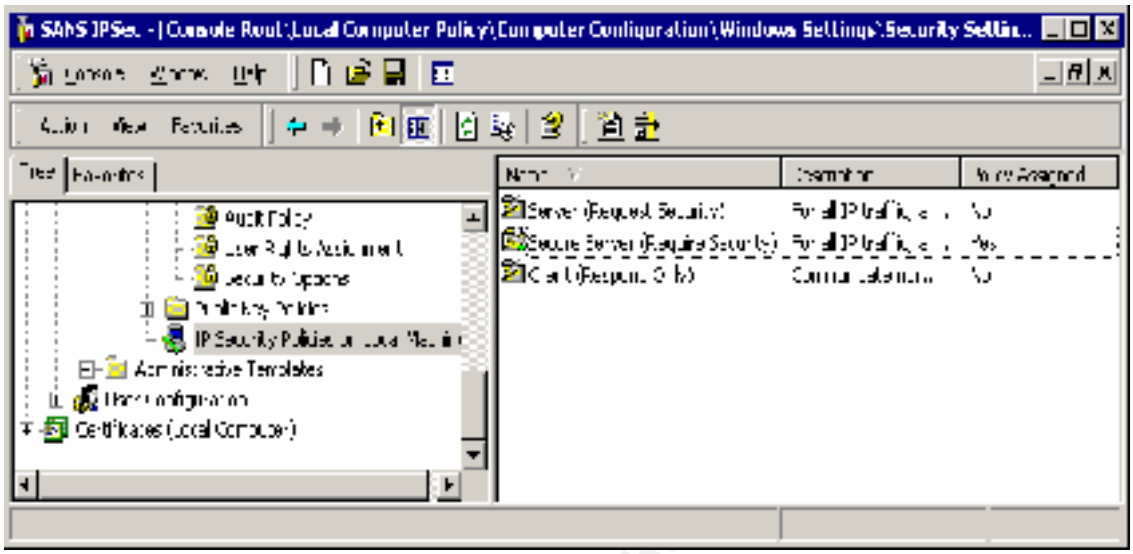
of the pre-defined policies is the Secure Server (Require Security) policy, which will not accept or send unsecured network traffic. Client hosts that try to communicate with a secure server must use the Server pre-defined policy or an equivalent custom policy.

Using the Secure Server IPSec Policy:

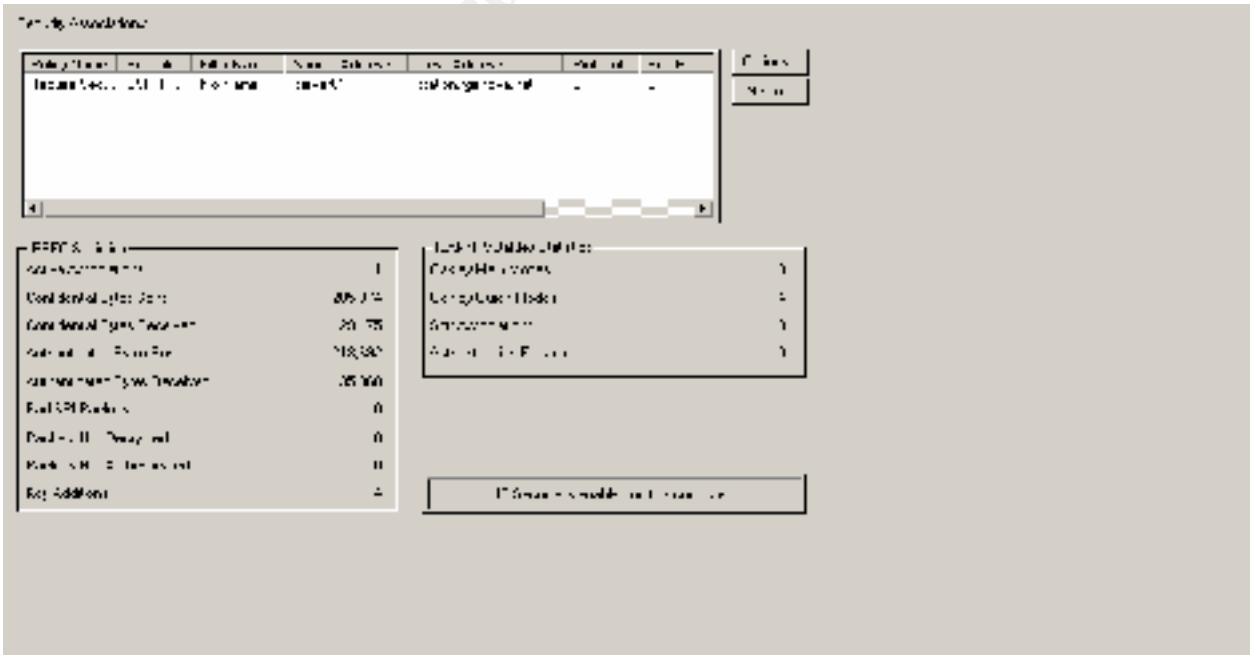
To demonstrate a Microsoft pre-defined IPSec Policy in this paper, I've chosen to use the highest level of security available by employing the Secure Server policy. Begin the selection of the Secure Server policy for **Server 01** by double-clicking on **IP Security Policies on Local Machine** in the left pane of the MMC we created for managing IPSec. Select **Secure Server** and the IP Security Rules are displayed. Note that all IP traffic requires security, while all ICMP traffic is permitted. Authentication is via Kerberos and no tunneling is applied.



The Secure Server policy demands that all network traffic to the host machine uses IPSec. This is what we want for that department or workgroup server with sensitive information. Secure Server policy will always request a secure channel and deny connections to computers unable to respond to the demand. You must also realize that since the security policies are bi-directional, if a Secure Server attempts to connect with a non-IPSec server (e.g., DNS or DHCP servers), the connection will fail. You will not, for example, be able to reach the Internet from the Secure Server desktop if your DNS resolution is provided by a non-IPSec-aware gateway server. The policy is not implemented until it is **Assigned.** To assign the **Secure Server** policy to Server 01, right click on **Secure Server** in the MMC and choose **Assign**. The **Policy Status** column should change from **No** to **Yes**.



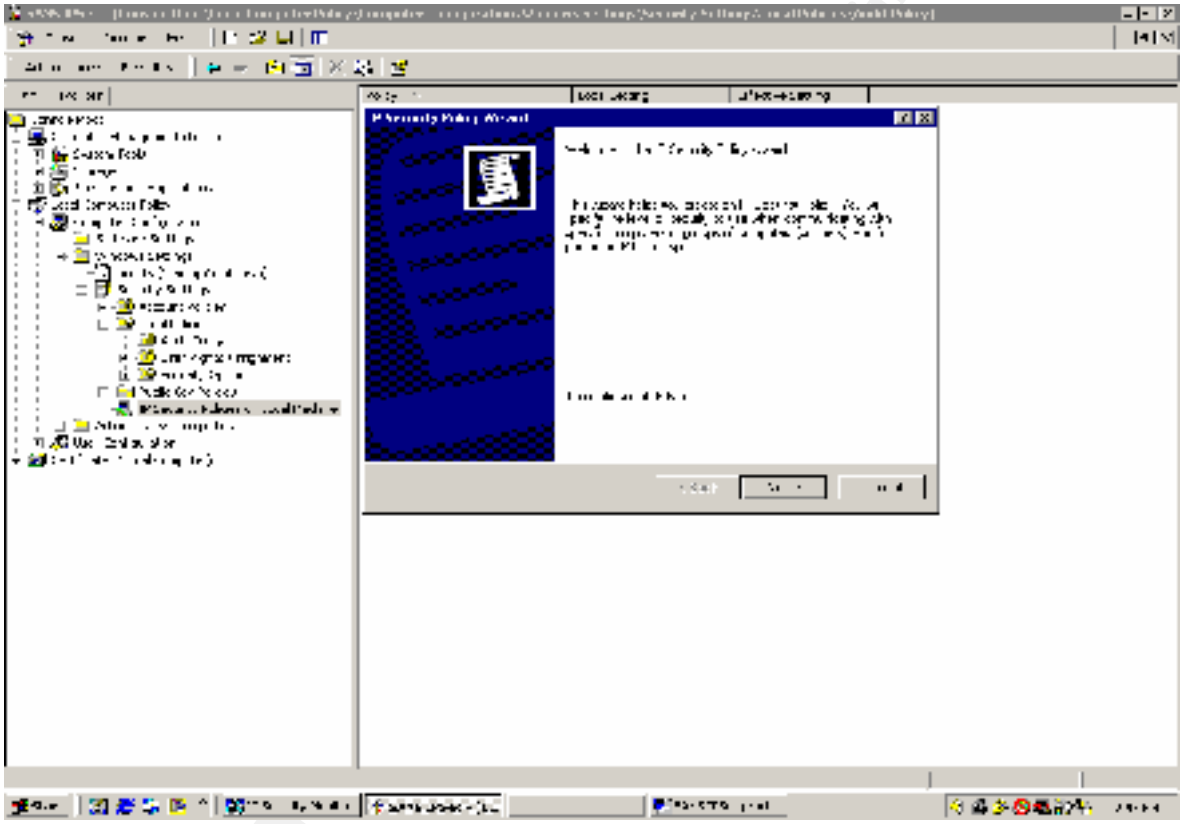
On the member server in the **rgenova.net domain**, the same steps for creating the IPSec policy on **Server01** were applied for **station.rgenova.net,** creating another **Secure Server** host. In this demonstration we'll have our two secured servers, **Server01** and **Station,** negotiate a session**.** To test our IPSec policies on the two computers, I sent an ICMP packet by pinging to Station at 192.168.1.33 from Server 01 at 192.168.1.99. Maximize the IPSec Monitor window created earlier and observe in the Security Associations and IPSec Statistics window that the source ping from Server01 successfully returned an echo packet from Station.



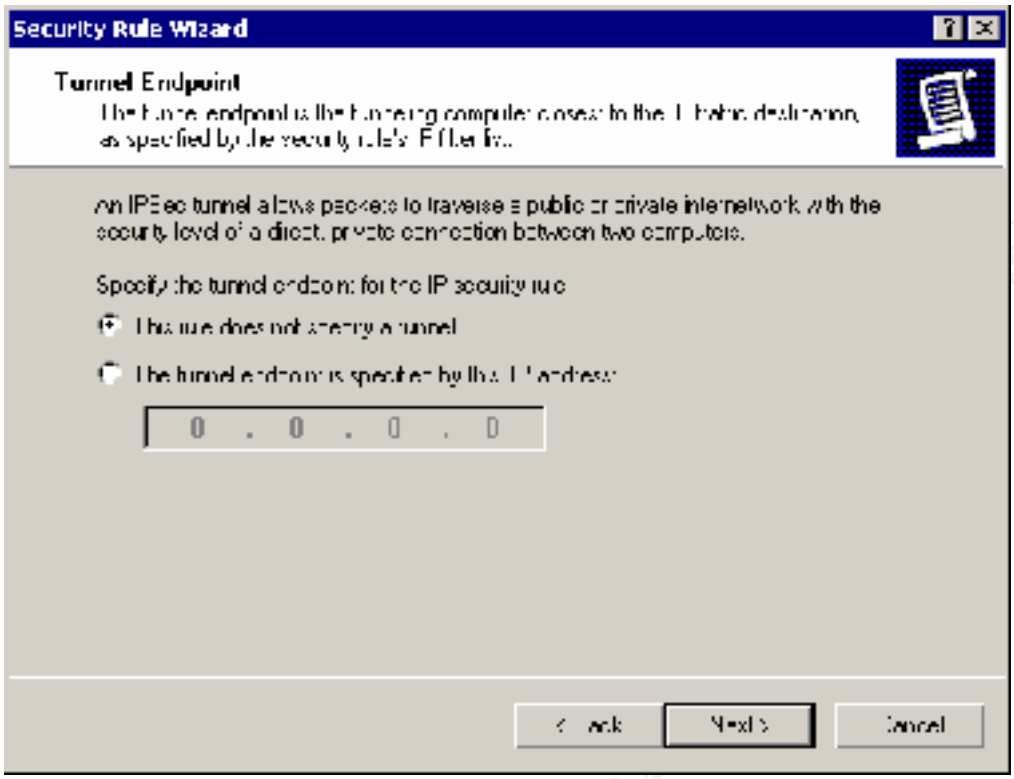
Launch the MMC on Server01 and click the + next to **Computer Management**, then expand **System Tools** and expand **Event Viewer**, followed by clicking on the **Security Log**. Double click the top instance of **Success Audit** to view the IPSec SA.

7. Custom Policy Editing

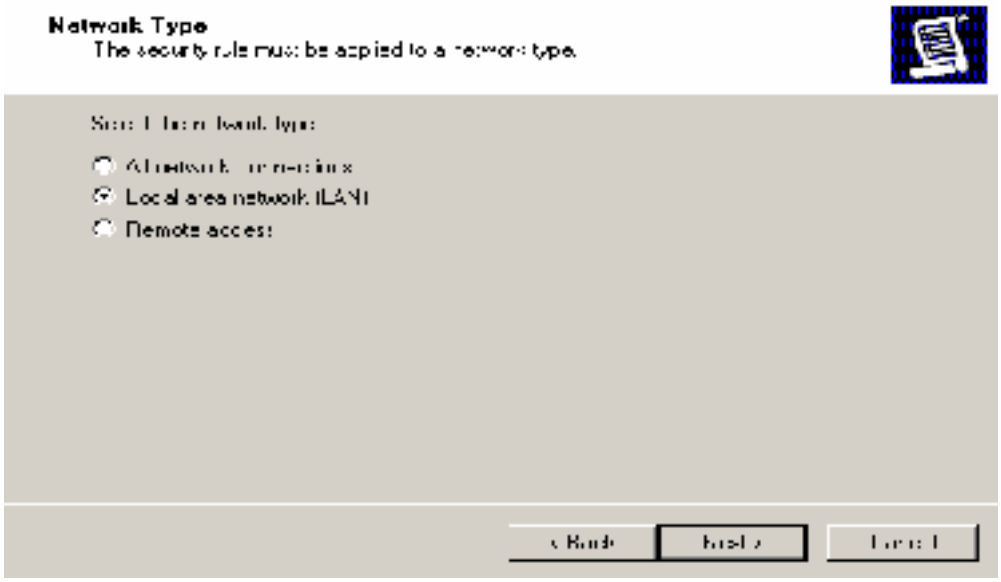
To create a custom policy, the first step is to launch the MMC and select IP Security Policies on Local Machine in the left pane. Select **Computer Configuration** and expand **Windows Settings**. Right click on **IP Security Policies** and click on **Create IP Security Policy**. The **"IP Security Policy Wizard**" will appear--click Next.



Enter a name and description for the new Security Policy. In the next step, clear the **Activate the default response rule** check box, then click finish. (If you don't clear the **Activate the Default Response Rule** box, you'll see a dialog box for choosing the authentication method. Kerberos version 5 is the default, but it is only allowed on machines that are domain members). In the next dialog box, check the **Edit Properties** box. In the **Properties** dialog box for the new custom policy, check that the **Use Add Wizard** box in the lower-right corner is defaulted, and click **Add** to start the **Security Rule Wizard**. The wizard will ask if you want to specify a tunnel endpoint. If you are running a DNS service on the network you can specify that, or the endpoint could be an IP address.

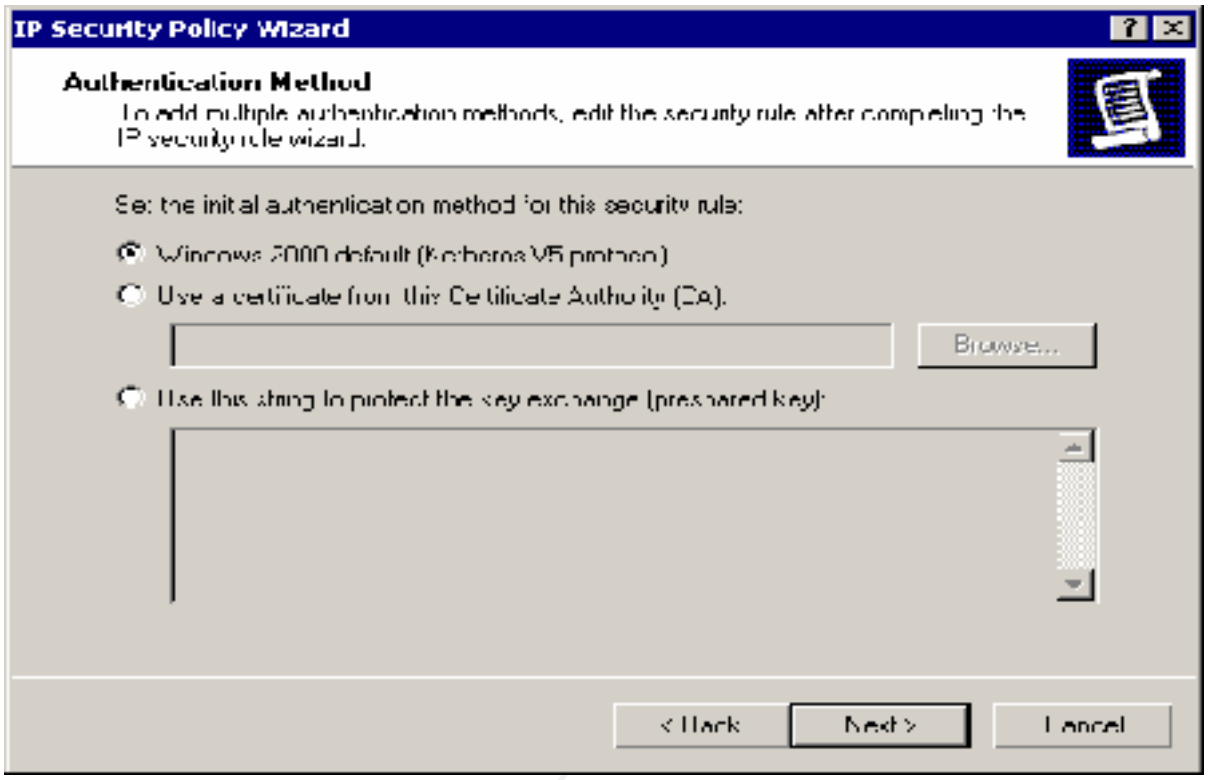


The next step in the wizard allows you to refine a rule based on connection type. The default is **All Network Connections**. If you were creating your policy only for the LAN or a Remote Access machine you could create a stricter rule by changing the default radio button here.



After the **Network Type** is specified, you must specify the IKE authentication method for a trust with a remote computer. The default method is Kerberos V5 Protocol and if you are

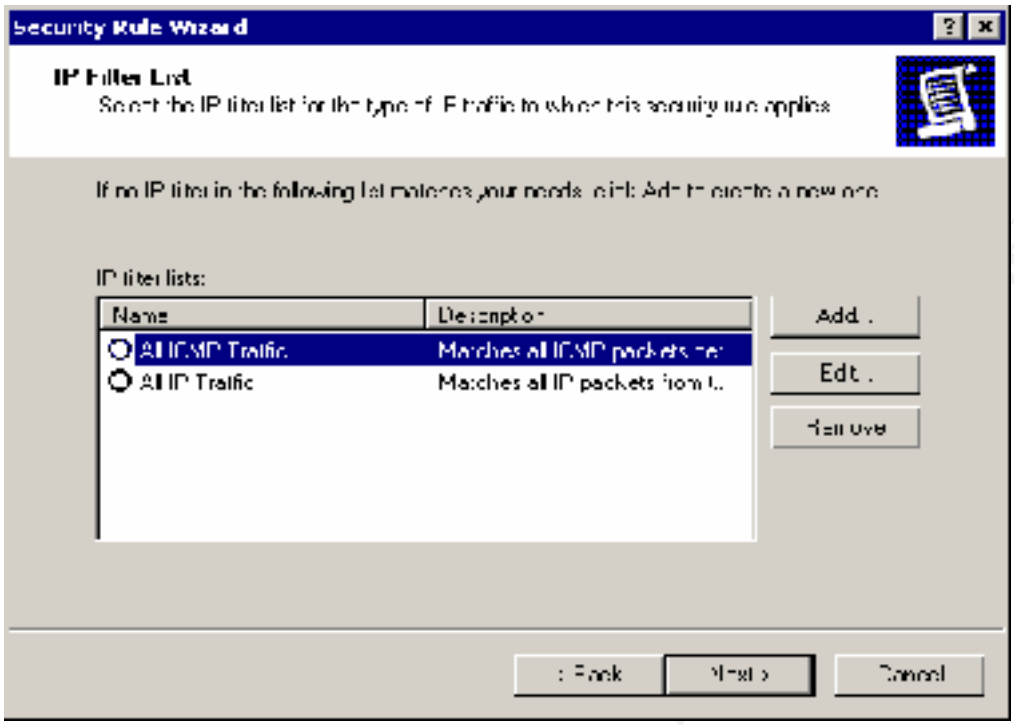
working within a Windows 2000 domain this represents the easiest deployment option with strong security.



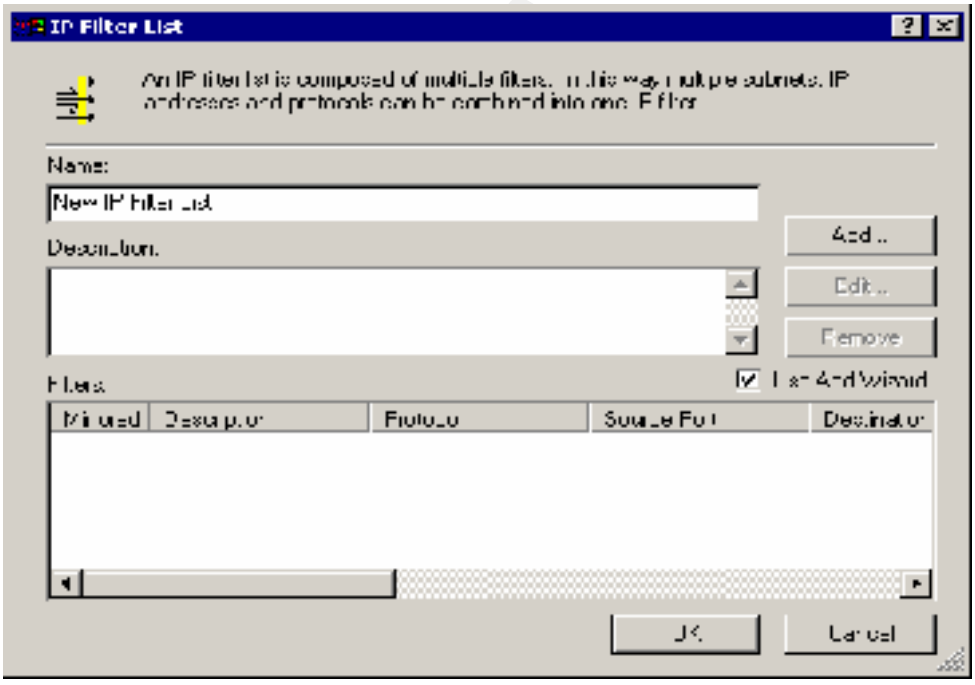
If there is external IPSec communication outside of the enterprise such as with an extranet, then the recommended option is to use a public key infrastructure by using certificates for authentication. This is easier to manage and represents much stronger security than the "preshared key." The shared secret authentication (password) method should only be used if there are interoperability issues that prevent the use of certificates or Kerberos. The shared secret is actually stored in plain text within the IPSec policy, so anyone with a valid domain user ID for a computer that is a member of the domain where the IPSec policy is stored in the AD can see the authentication key!

The final step in configuring a custom policy is creating the IPSec Filter List. Outbound packets are checked against filters to determine whether they should be secured, blocked or passed through (clear text). On the inbound side packets are also checked to determine if they should secured, blocked or passed into the internal network. IP filters are always mirrored for secured traffic. Mirroring means that the filters automatically configure both inbound and outbound traffic according to the filter list.

In the console for the **Security Rule** wizard, select **IP Filter List** and click add to display an empty list of IP filters. Name your filter and select the **Use Add Wizard** and then click **Add** again to start the **IP Filter Wizard.**



To create a new IP filter, click Add. For example, if you had a DMZ web server you could write a rule that allowed only HTTP and HTTPS traffic by blocking all ports except 80 and 443 respectively and thus allowing ICMP packets to pass.



In a filter list you would create two filters for such a policy with the following values:

Filter 1
Source Address—Any IP address
Destination Address—My IP address
Mirrored—Yes
Protocol—TCP
From Port—Any
To Port—80

Filter 2
Source Address—Any IP address
Destination Address—My IP address
Mirrored—Yes
Protocol—TCP
From Port—Any
To Port—443

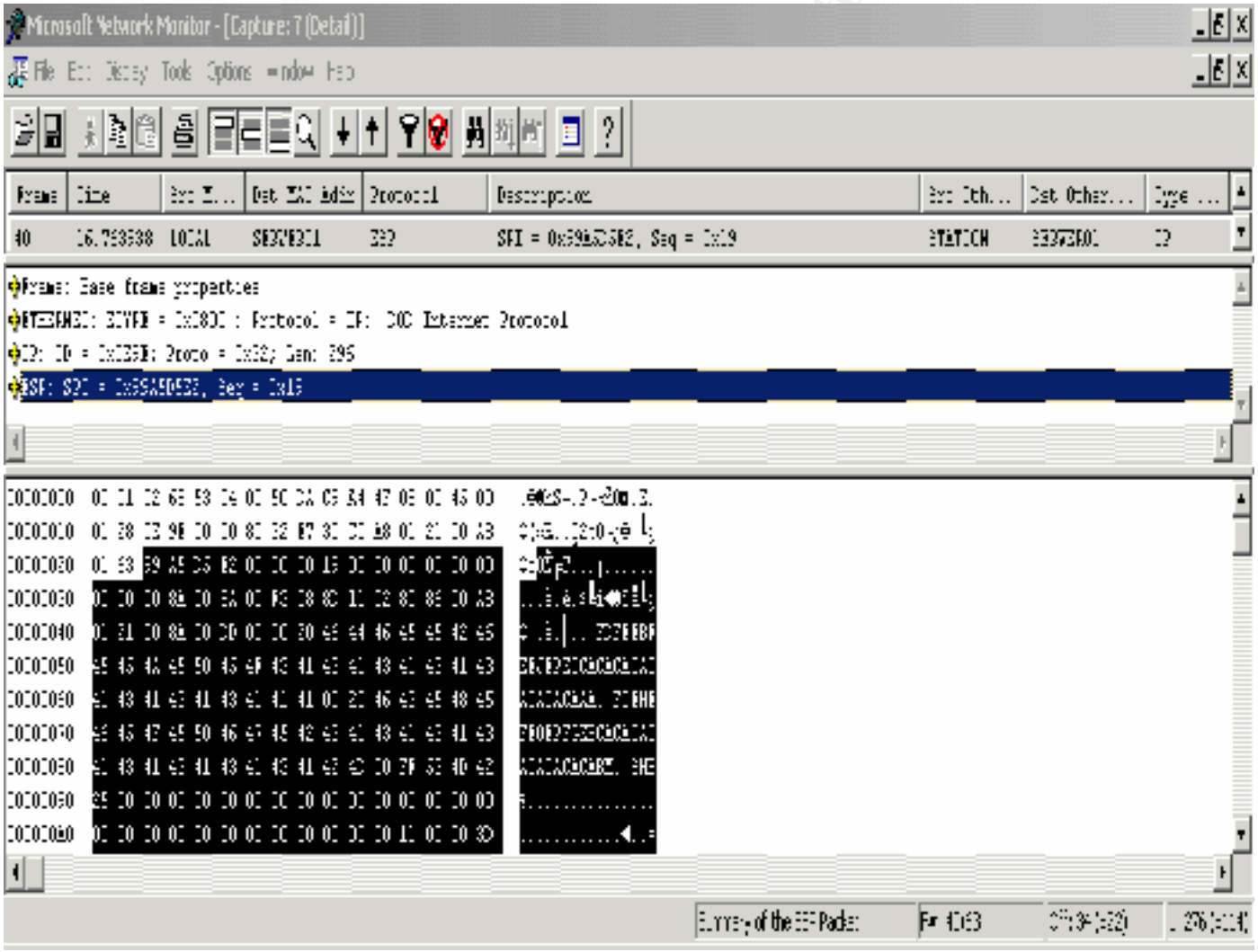
To write such a rule try using "Ipsecpol.exe," the Windows 2000 Resource Kit scripting tool for IPSec, rather than the graphical IP Filter List.* (Ipsecpol is also available from Microsoft at http://www.microsoft.com/Downloads/Release.asp?ReleaseID=19889 by extracting it from the IIS Lock.exe download). Below is a sample script for securing the DMZ web server's ports except for HTTP and HTTPS traffic. The commands below will create an IPSec policy called "WebFilter" that blocks all protocols to and from the DMZ web server and all other hosts; "OkHTTP" and "OkHTTPS" permits traffic on port 80 and port 443 to and from the DMZ web server and all other hosts. ICMP traffic including pings is blocked with this "WebFilter" rule.**

```
ipsecpol \\computername –w REG –p "WebFilter" –o
ipsecpol \\computername –x –w REG –p "WebFilter" –r "BlockAll" –n BLOCK –f 0+*
ipsecpol \\computername –x –w REG –p "WebFilter" –r "OkHTTP" –n PASS –f 0:80+*::TCP
ipsecpol \\computername –x –w REG –p "WebFilter" –r "OkHTTPS" –n PASS –f 0:443+*::TCP
ipsecpol \\computername –x –w REG –p "WebFilter" –r "OkICMP" –n PASS –f 0+*::ICMP
```

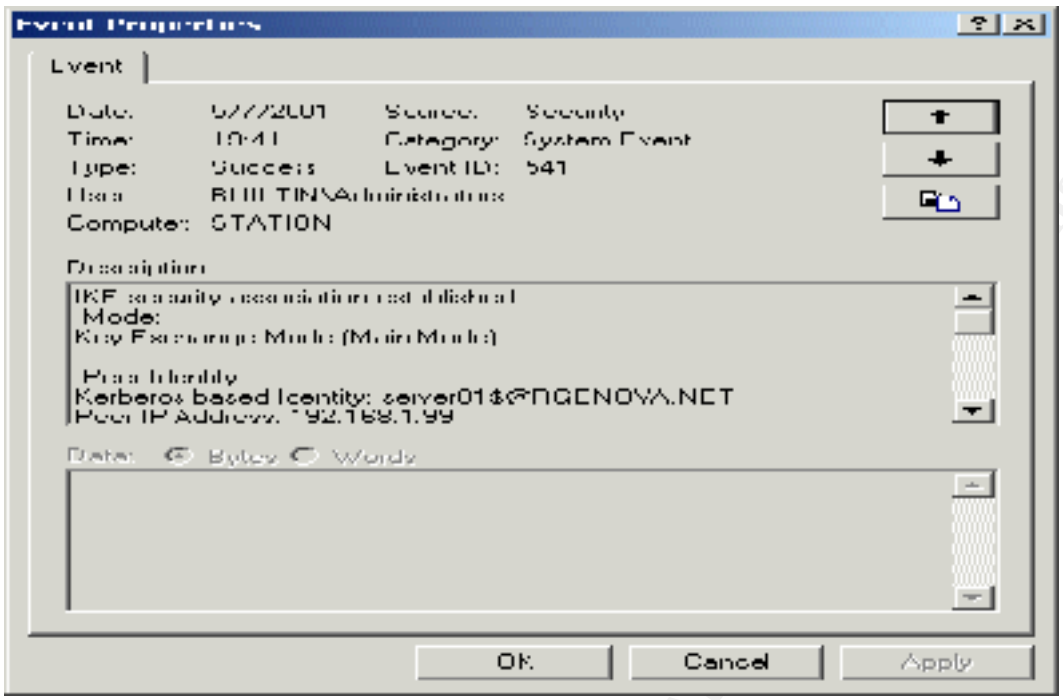
An important caveat for writing IPSec rules whether you use the graphical mode or the scripting tool is that by default IPSec filters do NOT block port 500 (UDP) nor do they block port 88 (TCP/UDP) on Windows 2000 domain controllers. These are used for IKE and Kerberos authentication respectively. However, in Service Pack 1 for Windows 2000, Microsoft has a feature to filter port 88. Add the DWORD registry subkey NoDefaultExempt to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IPSec with a value of 1.

8. Troubleshooting and Monitoring

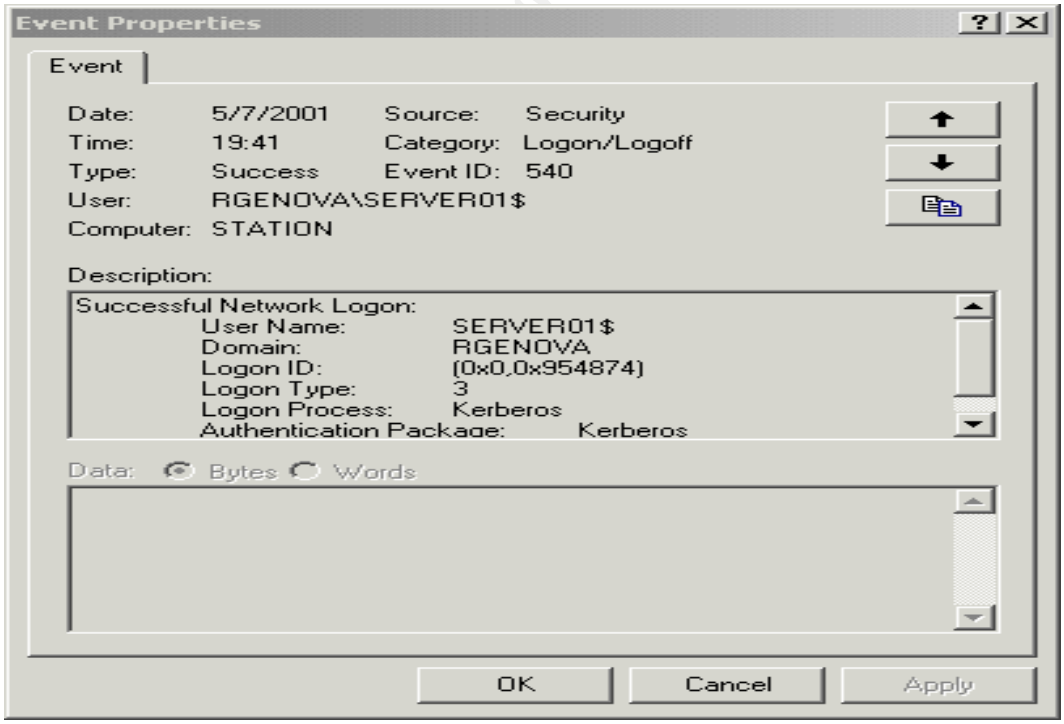
IPSecmon was demonstrated earlier during the Secure Server Policy installation to verify that IPSec data traffic was transmitted and encrypted. Now we'll consider some other monitoring and troubleshooting tools and techniques to employ for investigating IPSec traffic. Since this is Windows 2000, you can use Network Monitor version 2.0 which has parsers for IPSec and ISAKMP packets as a troubleshooting tool. Because Network Monitor sniffs the packets after IPSec, an encrypted packet will not display its contents, but its transmission over a network interface will be visible.



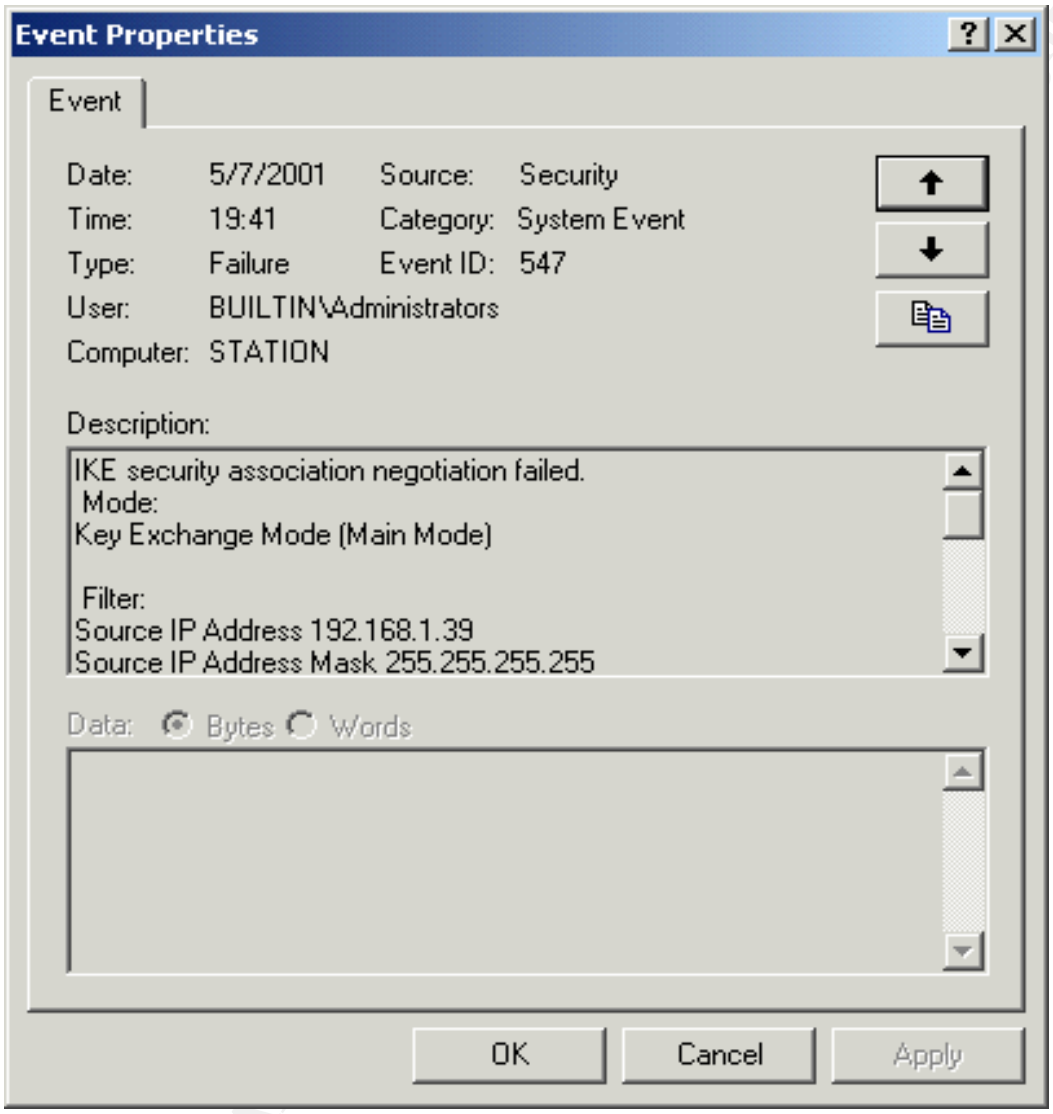
Event Viewer can be used to record policy agent and IPSec driver events in the system log. IKE/Oakley events can be recorded in the application log, and if logon auditing is enabled ISAKMP events with SA details will be recorded in the security log.



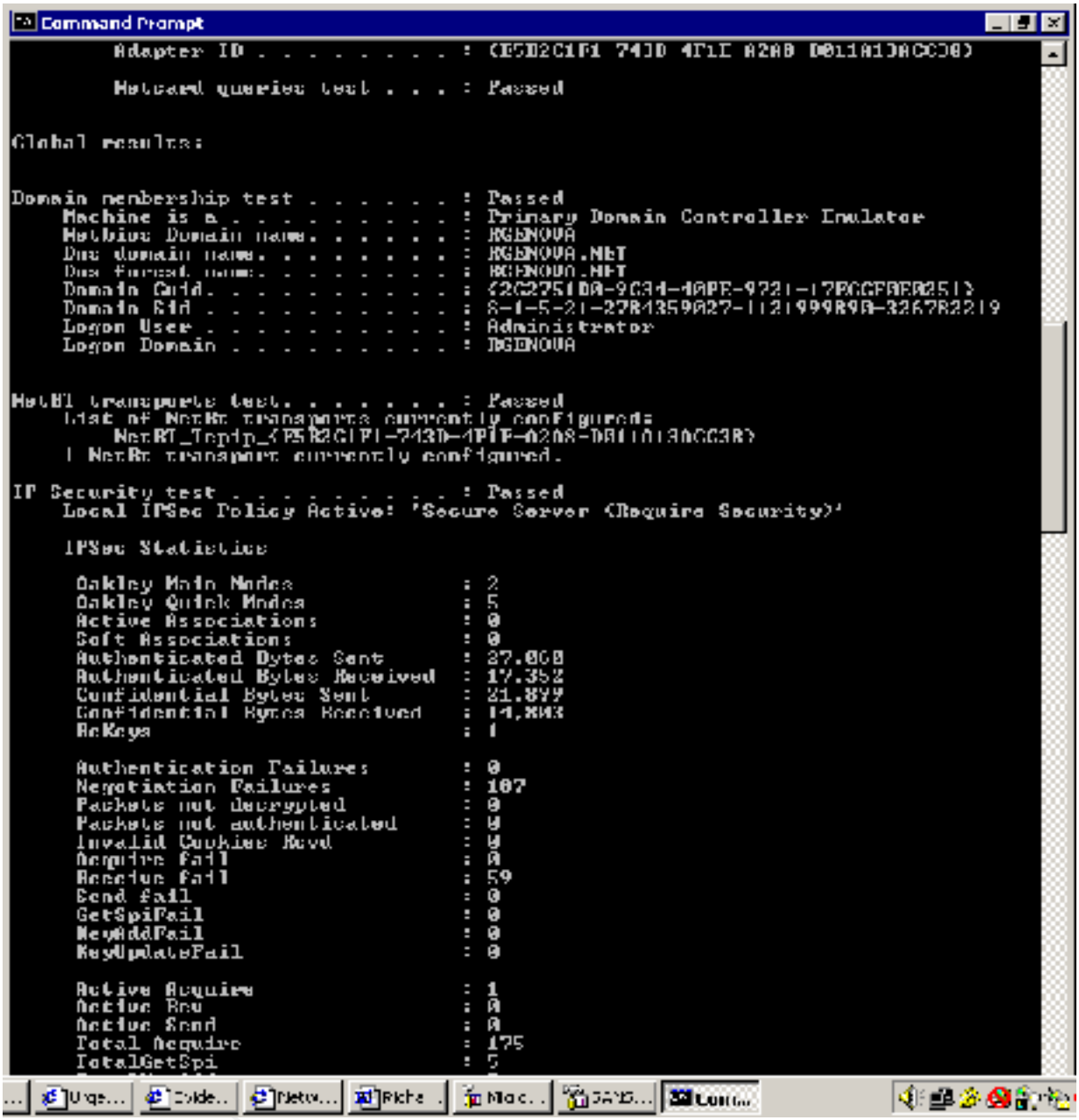
The detail from the event log above shows a successful Kerberos authentication and key exchange between an Administrator of Server01 (192.168.1.99) and Station (192.168.1.33). Below is another detail from the same session showing the logon success identifying user RGENOVA.



Problems will also appear in the event log. Below is a failed IKE negotiation between Server01 and Station which was changed to a multihomed machine with two network cards (notice the new IP address 192.168.1.39). IPSec in Windows 2000 will not succeed on all adapter interfaces in a multihomed computer!



Another way to monitor and troubleshoot failures and successes of IKE negotiation besides Ipsecmon and the Security event log is to use the command line support tool **netdiag.exe /test:ipsec /v**. The **Netdiag** executable is in the \Support folder on Windows 2000 Professional and Server CD-ROMs. Note the "Negotiation Failures : 187" under IPSec Statistics in the screenshot below reflecting the failure to establish an IKE Security Association between Server01 and Station.



If you need to clear IKE negotiations that are hung, you must stop and start the policy agent service from a command shell prompt and you must be logged on as a local administrator. Use "**net stop policyagent**" and "**net start policyagent**" to restart, then try again to secure traffic. If you were running the Routing and Remote Access service or had incoming VPN connections when you stopped the policyagent, then you must stop and restart the remote access service to reestablish IPSec protection. After restarting the IPSec policyagent, use "**net start remoteaccess."**

This has been far from an exhaustive presentation of the troubleshooting tricks and methods for IPSec, but it should help you to begin you're root cause analysis and to understand what to look for in IPSec data traffic captures.

9. Planning an Effective IPSec Implementation in Windows 2000

It's time to reflect on what constitutes an effective implementation of IPSec. Arguably this section could have been placed earlier, but I wanted the readers who are presumably IT professionals to experience the power of IPSec. Now that you have done so, let's look at a number of basic questions, many of which are common to any sound approach to security design, not just Windows, to consider during the earliest planning stages:

What are you protecting? What are the vulnerabilities?

- **Evaluate the type of information being sent over your network.** Is it sensitive financial data, proprietary information, or electronic mail? Because of their function, some departments may require a higher level of security for their data than does the majority of the enterprise.
- **Determine where your information is stored, how it is routed through the network, and from what computers it will be accessed**. This provides information about the speed, capacity and utilization of the network prior to implementation, which is helpful for performance optimization.
- **Evaluate your vulnerability to network attacks.**
- **Design and document an enterprise-wide network security plan.** Take into account the security framework of Windows, including the Active Directory model, and how security is applied to Group Policy.

Consider:

- **What should you secure?** Should you secure traffic between some computers or all computers, or only certain protocols or ports?
- **How should you secure it?** Should you secure the traffic with integrity only or confidentiality also, and at what strength? (For confidentiality you'll have use AH and ESP.)
- **Where should you secure it?** Should you secure it just over remote access connections, or also through the local area network?
- **Who will manage policy?** Should domain administrators, server administrators, or local computer administrators?
- **Will encryption settings work with all pertinent computers?** Will data be accessed by computers using strong cryptography (3DES encryption) as well as those with standard cryptography?
- **Will network and application performance parameters be affected?** Are there configuration or resource issues that need to be addressed to prevent introducing unacceptable network latency or performance bottlenecks? Will more memory or other resources be needed to handle the additional overhead introduced? Should hardware or software accelerators be deployed to offload encryption tasks?
- **Design, create and test the IPSec policies, to clarify and refine what policies and policy structures are truly necessary**. During testing of your deployment scenarios, run normal workloads on applications to gain realistic feedback. During initial tests, if you

want to view the packet contents with Network Monitor or a sniffer, use the **Medium** security method level or a custom security method set to **AH**, since using **High** or **ESP** will prevent viewing of the packet. We will start with predefined policies to demonstrate IPSec in action and to ease configuration.

- **Reduce administrative overhead spent on policy by using the predefined policies, rules and filter actions whenever possible**. They can be activated, modified, or used as a template for defining your own. (For more info about IPSec filters, please refer to http://www.windows.com/windows2000/en/datacenter/help/sag_IPSecbpspecial.htm.)

10. Summary and Conclusions

For guidance in determining when it's appropriate to implement Point to Point Tunneling Protocol, Layer Two Tunneling Protocol or IPSec with Windows 2000, I've reproduced a Microsoft chart outlining which security protocol to use or to combine based on differences in feature sets:

Feature	Description	PPTP/ PPP	L2TP/ PPP	L2TP/ IPSec	IPSec Xport	IPSec Tunnel
User Authentication	Can authenticate the user that is initiating the communications.	Yes	Yes	Yes	WIP[1]	WIP
Machine Authentication	Authenticates the machines involved in the communications.	Yes[2]	Yes	Yes	Yes	Yes
NAT Capable	Can pass through Network Address Translators to hide one or both end-points of the communications.	Yes	Yes	No	No	No
Multiprotocol Support	Defines a standard method for carrying IP and non-IP traffic.	Yes	Yes	Yes	No	WIP
Dynamic Tunnel IP Address Assignment	Defines a standard way to negotiate an IP address for the tunneled part of the communications. Important so that returned packets are routed back through the same session rather than through a non-tunneled and unsecured path and to eliminate static, manual end-system configuration.	Yes	Yes	Yes	N/A	WIP
Encryption	Can encrypt traffic it carries.	Yes	Yes	Yes	Yes	Yes
Uses PKI	Can use PKI to implement encryption and/or authentication.	Yes	Yes	Yes	Yes	Yes
Packet Authenticity	Provides an authenticity method to ensure packet content is not changed in transit.	No	No	Yes	Yes	Yes
Multicast support	Can carry IP multicast traffic in addition to IP unicast traffic.	Yes	Yes	Yes	No	Yes

Source: Microsoft White Paper, "Microsoft Privacy Protected Network Access: Virtual Private Netowrking and Intranet Security, 1999

[1] Support is not yet provided; however, there is work in progress (WIP) by the IETF IPSec working group.
[2] When used as a client VPN connection, it authenticates the user, not the computer. When used as a gateway-to-gateway connection, the computer is assigned a user ID and is authenticated.

The example implementations of IPSec technology under Windows 2000 in this practical paper only scratch the surface in understanding the power of this security tool. There are also some important specific configuration limitations and potential problems that must be noted about IPSec in Windows 2000 to conclude this practical.

a) *Performance Considerations*

Please consider the question of performance carefully if you deploy IPSec, because the encryption and decryption processes burn up a lot of CPU cycles. In a corporate production environment you would naturally first install IPSec policies on test servers and record before-and-after baseline performance statistics. Microsoft's well-known PerfMon tool that is included with Windows 2000 (and the Server Monitor tool in Internet Security Acceleration Server 2000, if you are deploying that product as well), can be used to obtain the desired metrics. Of course, you will discover that IPSec has an impact on CPU load and network latency because IPSec use will increase processor utilization while increasing IP traffic and IP packet sizes. Depending on the number of concurrent users you have (or simulate), and your choice of protocol and encryption algorithms employed, the CPU load on your servers could grow by as much as 90% when IPSec is used. A worst case scenario in terms of performance would involve using ESP with 3DES encryption with many clients.

There are two ways to address the performance hits to manage them to acceptable levels. One is to use extra processors to distribute the CPU load and the other is to offload the encryption/decryption process to network interface cards that can handle the computational stress. Intel, 3COM and other manufacturers are marketing NICs with this capability. Using a multiprocessor server with NICs that utilize their own embedded encryption processors can reduce significantly the network and server performance degradation. For example, 3COM states in their product literature that its "trademarked 3XP Processor shares a greater load of network traffic processing" and "secures sensitive data by delivering 3DES, DES, MD5, and SHA-1" encryption. 3COM claims a "25 to 35 percent savings running cycle-intensive IPSec tasks" and that their specially equipped NIC "provides the necessary memory to support up to 1024 IPSec Security Associations." Intel makes similar claims. (You may have noticed the use of 3COM network cards in my screen shots, but no endorsement or verification of specific claims made by any NIC manufacturer is made or implied by this author.) With your own baseline numbers, you can evaluate the accuracy of vendor claims, but there is no doubt that a combination of multiprocessing power and enhanced NICs will be necessary in a large corporate deployment.

b) Network Address Translation and Firewalls

Two likely thorny issues for network administrators who are responsible for securing communications for telecommuters or remote clients with broadband access to the corporate network are Network Address Translation (NAT) and communicating through firewalls. Because the NAT process translates the original source IP address of packets passing through the NAT service to a common IP address while the source port is translated to the actual port established by the NAT server, an IPSec protected packet won't be translated without invalidating the packet if it's AH-protected or has ESP-protected data. This knocks out the possibility of using IPSec with broadband connections from a public to a private network if there is a NAT service sitting

between the two. Remote network users are rightly being encouraged or required by sound corporate policy to install rudimentary desktop or personal firewalls. These "consumer" firewalls, whether they are hardware- or software-based, usually rely on NAT for shielding the Internet user from hackers trawling for open ports on Internet-connected computers, but this is at odds with using IPSec as a VPN for telecommuting employees.

Similarly, implementing IPSec when network traffic must pass through firewalls introduces some important considerations. To pass IPSec datagrams through the firewall, you must allow UDP 500 packets and a protocol identifier (ID) of 51 for AH or a protocol ID of 50 for ESP pass-through (and the firewall must not be performing NAT). A big problem here is that IPSec using ESP may lead to a firewall losing its ability to perform stateful inspection of data. Making an exception in the firewall rules for UDP port 500 allows all ESP-protected data to pass and there's no way to determine which protocol is encrypted within the packet. This can allow unauthorized traffic to enter the network if the source and destination hosts establish their IPSec connection! Similarly, it is also very difficult to properly configure IPSec filters if an application uses random ports. This is a big problem with many web-enabled DCOM applications which use dynamic port mapping (especially bad from a security standpoint are those using the SunRPC port 111 as a portmapper). Targeting the well-known port 111 for exploits is a favorite of hackers.

If Microsoft's new Internet Security and Acceleration Server 2000 (ISA) is part of your network's security protection plan, you should be aware of how these potential pitfalls will affect achieving interoperability of IPSec and ISA. It's possible to enable IPSec on a machine running ISA server, if the ISA server is a VPN machine using Layer 2 Tunneling Protocol (L2TP). IPSec will be automatically used by L2TP for data encryption. Be aware, however, that when IPSec is enabled the AH and ESP protocols are controlled by the IPSec driver instead of ISA's packet filter driver. Therefore only valid AH and ESP traffic will be allowed to enter the network. If the ISA server is configured to block IP fragments, AH and ESP fragments will also be blocked even if IPSec is enabled on the server. And as explained above, NAT is incompatible with protocols that use the IP addresses in fields other than the standard header fields, so IPSec cannot be used **through** an ISA server. **You may only use IPSec to encrypt L2TP data traffic using the ISA server as a VPN endpoint.**

c) Exclusions

There are also a number of excluded protocols that can't be protected with IPSec. IP broadcast addresses and multicast addresses can't be secured because IPSec filters only for single recipients of packets. IKE is used to negotiate the SA between two hosts participating in an IPSec transmission, so you cannot encrypt the negotiation process of IPSec. The negotiation must take place using plaintext packets that define how subsequent packets will be protected. If your network is using Quality of Service (QoS), IPSec can protect the protocol for which RSVP is requesting the QoS, but it cannot protect the RSVP packets used to request QoS settings. One other configuration (and potential legal) issue for network and system administrators using IPSec involves applying encryption protocols and integrity algorithms. **To use 3DES, the Windows 2000 High Encryption Pack must be installed and export restrictions must be considered if international network traffic involves restricted countries.**

To get started with IPSec and to get the most value from it, build and test the simplest end-to-end policy. Begin with pre-built, pre-defined policies provided by Microsoft. This will help you to avoid interoperability problems and troubleshooting nightmares. Proper use of IPSec can give you data privacy, integrity, authenticity, and anti-replay protection for network traffic end-to-end from client-to-server, server-to-server, and client-to-client in transport mode. For DHCP, DNS, WINS or domain controller servers, determine if all the clients support IPSec. Otherwise, if your IPSec policy is not configured to allow fall back to unsecured traffic for older clients, then secure negotiation might fail and access to these network services might be blocked. Again, the best and easiest deployment will be in a pure Windows environment with no downlevel clients.

It wasn't discussed in detail here, but IPSec can theoretically provide secure gateway-to-gateway connections across private WAN or Internet-based connections using Layer 2 Tunnel Protocol/IPSec tunnels **or pure IPSec tunnel mode**. However, Microsoft counsels that, "although IPSec in tunnel mode can be used alone to support remote access, the work toward a pure IPSec VPN is still in progress. The most significant issues currently are interoperability with different vendor implementations, and the inability to tunnel multicast and broadcast traffic. The latter hinders the ability to create router-to-router VPN connections using IPSec tunnel mode. **The solution for now is to use L2TP/IPSec for tunneling remote access connections**." (N.B. IPSec tunnel mode should not be used for VPN remote access.)

There is a constantly growing wealth of published material on Windows 2000 and IPSec, some of which is cited here in the compilation of sources at the end of this document, that will help you progress with your IPSec implementation. Not surprisingly, one of the best sources of IPSec information is Microsoft Technet. Excellent documentation for IPSec deployment is also available on the Windows 2000 Resource Kit CDs.

Sources and References

Baker, Doris M. and Mel, H.X., Cryptography Decrypted, Addison-Wesley, 2001

Blaze, Matt; Ioannidis, John; and Keromytis, Angelos, "Trust Management for IPSec," 1999

Crawford, Sharon and Russel, Charlie, Microsoft Windows 2000 Server Administrator's Companion, Microsoft Press, 2000

Dutcher, Bill, The NAT Handbook, Wiley, 2001

Ferguson, Niels and Schneier, Bruce, "A Cryptographic Evaluation of IPSec," 1999, http://www.counterpane.com

Hollandsworth, Jon, "Overview of IPSEC Manageability and Security," 25 July 2000, SANS Institute Information Security Reading Room

Howard, Michael, "Defense In-depth, Using IPSec Effectively in Windows 2000," 23 January 2001, http://www.secadministrator.com

Microsoft Technet, "Step-by-Step Guide to Internet Protocol Security," 17 February 2000, http://www.microsoft.com/TechNet/win2000/win2ksrv/technote/ispstep.asp

Schmidt, Jeff, Microsoft Windows 2000 Security Handbook, Que, 2000

Schultz, E. Eugene, Windows NT/2000 Network Security, New Riders, 2000

Last Updated: September 21st, 2017

Upcoming Training

Secure DevOps Summit & Training	**Denver, CO**	**Oct 10, 2017 - Oct 17, 2017**	**Live Event**
San Francisco Winter 2017 - SEC505: Securing Windows and PowerShell Automation	**San Francisco, CA**	**Nov 27, 2017 - Dec 02, 2017**	**vLive**
SANS San Francisco Winter 2017	**San Francisco, CA**	**Nov 27, 2017 - Dec 02, 2017**	**Live Event**
SANS Cyber Defense Initiative 2017	**Washington, DC**	**Dec 12, 2017 - Dec 19, 2017**	**Live Event**
SANS Southern California- Anaheim 2018	**Anaheim, CA**	**Feb 12, 2018 - Feb 17, 2018**	**Live Event**
Southern California- Anaheim 2018 - SEC505: Securing Windows and PowerShell Automation	**Anaheim, CA**	**Feb 12, 2018 - Feb 17, 2018**	**vLive**
SANS 2018	**Orlando, FL**	**Apr 03, 2018 - Apr 10, 2018**	**Live Event**
SANS OnDemand	**Online**	**Anytime**	**Self Paced**
SANS SelfStudy	**Books & MP3s Only**	**Anytime**	**Self Paced**