Global Information Assurance Certification Paper

GSEC Practical Assignment

Version 1.4b option 2

Case Study: Creating an IPSEC secure tunnel between a Symantec Enterprise Firewall and a Symantec Velociraptor appliance

By Michael Socher

Abstract:

The purpose of this paper is to explain the process of creating a site - to- site IPSEC VPN connection between a Symantec Enterprise Firewall and a remote municipality's Symantec Velociraptor appliance. More specifically, Symantec Enterprise Firewall version 7.0 running on Windows 2000 w/sp2 and a Symantec Velociraptor 1100 version 1.5 running on Linux. The current state of both the VelociRaptor and the Symantec Enterprise firewall are fully patched to the recommendations of Symantec Enterprises. At first glance of the configuration of the SEF version 7.0 running on Windows 2000 service pack two one may wonder why is service pack two being used when there are more recent service pack versions available? The answer to this posing question is that version 7.0, or more specifically 7.04 (Patched version of 7.0), is not compatible with service pack releases beyond service pack two at this time. Failure to comply with the service pack recommendations will result in an ill-functioning firewall. The version 7.0 and 7.04 software provides hardening to the Windows OS, which ensures that the firewall itself is secure. To begin I will explain VPN as a general "What is" along with the purpose for the project and how my working knowledge helped me in the decisions that were made. Next, I will explain the "nuts and bolts" of the project itself. This will include the steps taken to complete the working secure tunnel. Finally, to conclude, I will explain the success of the project and how the municipality has improved productivity through the access of the secure tunnel.

VPN introduction:

First of all I will try not to wonder too far into explanation on the inter operation of IPSEC and VPN. I will assume readers have some sort of knowledge of the concepts that make site-to-site VPN possible. With that said, let's get into some definition.

The Virtual Private Network Consortium defines VPN as (7):

- A *virtual private network* (VPN) is a private data network that makes use of the public telecommunication infrastructure, maintaining privacy through the use of tunneling protocol and security procedures. A virtual private network can be contrasted with a system of owned or leased lines that can only be used by one company. The main purpose of a VPN is to give the company the same capabilities as private leased lines at much lower cost by using the shared public infrastructure. Phone companies have provided private shared resources for voice messages for over a decade. A virtual private network makes it possible to have the same protected sharing of public resources for data. Companies today are looking at using a private virtual network for both extranets and wide-area Intranets.

A more general description of VPN would be a network that can safely be used as if it were private or internal, even though some of its communication uses insecure connections. All traffic on VPN connections is encrypted. Furthermore, data sent over a VPN tunnel to selected endpoints works by packaging together and encrypting data as well as an authenticating packet. Once this entire package, if you will, reaches its destination, the subsequent authentication is removed to reveal the actual data to its intended recipient. This is what creates the possibilities of extended secure networks. Keep in mind that this paper is focusing on site-to-site IPSEC VPN but I would like to add that this technology also includes client-to-site. Not only do you have the option of an entire extended secure network but also you have the choice of a single remote node. To include, the completed tunnels can be controlled by the creation of granular rules of access between the endpoints. I will offer an explanation and demonstration of the rule creation later on in this paper. From the definition and further explanation above, the next spark of enlightenment that should come to mind other than security should be cost savings and the elimination of dedicated lines. This could mean that in some cases, a given IT department may have more money to spread to other projects. Cost savings is not the focus for this paper but it is a deciding factor for almost all IT projects.

The Project Purpose:

Many organizations today survive through multiple locations that all need some way to securely transfer data between them. This is the case for this paper. A department within the organization for which I work recently came up with an idea to create a grouping of databases that various types of sensitive data would be stored. Access to the data would be through a web interface that requires a user name and password. Through this http connection, a user would be able to access criminal and non-criminal citizen records, casework, not excluding other forms of data. Within the criminal records and casework, there are numerous images that accompany the data. This data needed to be accessible to near by municipalities. From an internal standpoint this has not been a problem. Users simply connect to the intranet site from anywhere on the trusted network, enter access information, and then pull the requested data. The problem now is that how do provide secure access to an internal database with an internal authentication http type GUI (Graphical User Interface). Providing authentication through a publicly facing site is not an avenue that could be pursued for my organization.

A few ideas were thrown around about a connection solution. One solution is the use of private lines, which are not an option for this project due to the price. Another was to set up individual VPN client software on each of the remote nodes. The second solution would have proved to be management nightmare. If anyone has ever worked with setting up client VPN software on various platforms you will know what I'm talking about. To add, the first site alone would have had at least 30 to 40 nodes. This again is not an option. Another option that we decided against is a dial up solution that would connect in through

what we call a service network. This service network is nothing more than a private network that extends from our enterprise network. To access this network would involve installing individual clients that would dial in to a private number in order to authenticate with a RAS server. This solution only provides authentication and does not provide encryption. Not only did this solution not meet the level of security we were looking for but it also would have been mind numbing to transfer data over a dial up connection.

Site Firewall Solution:

Up until this point I did not like the way this project was trying to pan out. Luckily there is light at the end of the tunnel. Just to give a little more foundation, the explanation of this paper pertains to the first site that this solution was deployed to. The first deployment site at the time was running a Microsoft proxy server doubling as a firewall for their main line of perimeter defense. As with many products, interoperability can be an issue. The creation of a site-to-site VPN connection between a Microsoft Proxy server and a Symantec device is no exception. For this project to become a reality moving away from Microsoft proxy server and in to a more compatible solution is a preliminary step. This particular site had already planned for the deployment of a new firewall solution to replace the proxy at an earlier time. Through consultation and interoperability with our current Symantec Enterprise Firewall the remote site decided to go with a Symantec Velociraptor.

The VelociRaptor 1100 is an appliance-based solution that includes licensing for 100 concurrent inbound and outbound connections combined which is adequate for all sites that will participate in this project. The remote municipality also elected to release all administration and configuration rights until I can get them up to speed on training for this new firewall. This was decided way before the site-to-site VPN solution. Configuration of the sites Velociraptor also gave me a great understanding of the remote network, which brings me to configuration tools. With using Symantec products, management and configuration is simplified through the use of the SRMC or Symantec Raptor Management Console. Symantec defines this piece of software as “the graphical user interface used to configure the security gateway.” (9) This software allows you to do anything you may need to within the firewall whether it’s local or remote. Being a paranoid security analyst I verified that this configuration tool created an encrypted connection with Symantec engineers before proceeding. Now that the big picture has been presented we can move forward with this process.

VPN Policy decision:

Site-to-site VPN from my organizations standpoint will be the secure connection of choice over client-to-site. This is mainly due to the number of current and potential users of this data service my organization is providing. For this project to become a reality, I had to decide on the creation of a VPN policy and what components will be used within that policy. The Symantec firewall/VPN

product line offers three encapsulation protocols. As listed in the Symantec Enterprise Firewall configuration guide these choices are (9):

1. *Ipsec.Static*
2. *Ipsec/IKE*
3. *SwIPe*

From a general description standpoint, these three options can also be looked at as static, hence the name Ipsec.Static, dynamic using IKE "Internet key exchange" or swIPe. The third option offers the lowest encryption levels so I did not see the need to explore it any further. The first two offer better encryption levels within IPsec. IPsec/IKE offers the best security of the three because of a perpetually changing key. Deciding on which of the three to use can depend on many factors including security policy and data sensitivity. The remainder of this project will pertain to the selection of SHA1 with 3DES encryption. With every new character added to a story comes some explanation of that character. So without further adieu let's present a few new players. IPsec (Internet Protocol Security) is one of the key components that pull this process together and is an Internet standard for interconnected, secure networking devices and the main technology used in VPNs (Virtual Private Networks). As stated in [RFC 2401] (0):

- *IPsec* is designed to provide interoperable, high quality, cryptographically based security for IPv4 and IPv6. The set of security services offered includes access control, connectionless integrity, data origin authentication, protection against replays (a form of partial sequence integrity), confidentiality (encryption), and limited traffic flow confidentiality. These services are provided at the IP layer, offering protection for IP and/or upper layer protocols.

The basic idea of IPsec is to provide authentication and encryption at the IP level. This requires IKE (Internet key exchange) to open up the path for the services ESP and AH. One can think of IKE as a protocol that allows two parties to initiate secure communications that is safe from uninvited guests in order to share a secret key. AH (Authentication header) handles the packet authentication service or integrity checking and ESP (Encapsulation security payload) actually encrypts and authenticates the data. The Symantec products use both AH and ESP so all communications through the tunnel start by having the senders identity verified by AH. Next, ESP is used to allow for the "Tunnel" and data encryption. Remember that with Symantec products, you can choose to not use AH. However, if AH is chosen to be used with a data privacy algorithm (DES, 3DES, AES), both ESP and AH are applied to the packet. Due to my current configuration both AH and ESP are used. A good point might be on why you would need one or the other or both. As previously mentioned, AH provides authentication to the IP header. Some IP header fields can change when traveling over the Internet. Changing values can affect source predictability and

cannot be protected by AH. ESP can be used to provide a similar service but also adds confidentiality. The primary difference between these two protocols is the authentication provided to the extent of coverage. Furthermore, ESP can encrypt the header but only if it is part of an encapsulated field. An official description of these two protocols as stated in [RFC 2402 and RFC 2406] is as follows (3, 2).

- The *IP Authentication Header (AH)* is used to provide connectionless integrity and data origin authentication for IP datagrams (hereafter referred to as just "authentication"), and to provide protection against replays.

- *ESP* is used to provide confidentiality, data origin authentication, connectionless integrity, an anti-replay service (a form of partial sequence integrity), and limited traffic flow confidentiality.

The last two characters I will be adding to this portion of the story are SHA1 and 3DES. As stated in [FIPS 180-1 and FIPS 46-3] (1, 5):

- *Secure Hash Algorithm (SHA-1)* can be used to generate a condensed representation of a message called a message digest. The SHA-1 is required for use with the Digital Signature Algorithm (DSA) as specified in the Digital Signature Standard (DSS) and whenever a secure hash algorithm is required for Federal applications. Both the transmitter and intended receiver of a message in computing and verifying a digital signature uses the SHA-1.
- *3DES*: The ANSI X9.52 standard defines triple-DES encryption with keys k_1, k_2, k_3 as C=Ek3 (Dk1 (M))).
- E_k and D_k denote DES encryption and DES decryption, respectively, with the key k. This mode of encryption is sometimes referred to as DES-EDE. Another variant is DES-EEE, which consists of three consecutive encryptions. There are three keying options defined in ANSI X9.52 for DES-EDE:
    - The three keys k_1, k_2 and k_3 are independent.
    - k_1 and k_2 are independent, but $k_1 = k_3$.
    - $k_1 = k_2 = k_3$.

Did you get all of that? In short the data is passed through the DES algorithm and is encrypted three times, which makes it 3DES or “Triple” DES. As for SHA1, it is an algorithm that takes a message as input and produces a hash or, a fixed-length set of bits that depend on the message contents in a complex manner. What SHA1 does is make it extremely difficult for anyone to duplicate or to change a message without altering its hash. Now what do we do with all of this new information? We put it all together in a policy that will allow us to configure our secure tunnel. From the options supplied within the Symantec products I have chosen Ipsec/IKE mainly for two reasons. The first reason is for its

configuration flexibility and second for its dynamic key ability. Symantec places what it calls a “Lifetime Timeout” which indicates that a tunnel can exist for a set amount of minutes before it is “rekeyed.” The default seems to work out pretty well which is 480 minutes. On top of the level of encryption, this ever changing or dynamic keying option gives an even higher added level of security. In the following sections I will explain how all of this information comes together using Ipsec/IKE for a VPN policy that includes data integrity of SHA1 and data privacy of 3DES.

Why SHA1:

In respect to the hash algorithms available to the Symantec products, MD5 and SHA1 are used to authenticate packets. Depending on preference, one of each or a combination of both can be used. Keep in mind that the configuration for data integrity must be the same at both ends of the tunnel in order to work. Through testing, I found that individually each worked well and were consistent. I did find that a combination method produced inconsistencies with the tunnel. From my analysis of the firewall logs there appeared to have been a complication in synchronization between the preferences each system chose to use. To avoid future user complaints and to ensure tunnel reliability, I elected to streamline the configuration and go with the option of SHA1. In addition SHA1 is a slower algorithm than MD5 but it is more secure because it produces higher bit hash of the input. This added security helped further sway my decision. I any case, either hash function offered with the Symantec product is acceptable because both are secure. In order to find a string that hashes to a given value produced by either function would be beyond the computing power available to most.

Why 3DES:

For this project I wanted to use an encryption protocol that is secure as well as certified. Although Symantec products can provide encryption levels above 3DES, choosing higher encryption levels may slow down your systems more than you would like. Before my decision to use 3DES I looked to the minds of ICSA www.icsalabs.com. Tests were conducted with Symantec Enterprise Firewall by ICSA concerning site-to-site VPN. The encryption protocol tested against the system was 3DES. Also, SEF specs that were used match what CountyXYZ uses. These specs are Symantec Enterprise Firewall version 7.0 running on Windows 2000 with service pack 2. From the ICSA Certified Cryptography Product Lab Notes they state (4):

- *Statistical and analytical tests were conducted on several cipher texts. No patterns were observed. The cryptography used on the analyzed cipher texts meets the accepted standards for this level of encryption.*

With the conclusion of the lab results on the Symantec products, I decided that the use of 3DES encryption should suffice without having to

sacrifice too much processing power. Later testing of the data passing through the VPN tunnel shows that only encrypted data is observed. From the encrypted data capture, I was able to conclude myself that 3DES would be acceptable. For further testing I experimented with higher and lower levels of encryption. The alternative options available and compatible between both devices are DES and AES. DES with its 56 bit key cipher proved to be the quickest at accessing data records and image files. However, DES is no longer considered to be secure so I had to test other options. AES (Advanced Encryption Standard) uses a completely independent algorithm from DES and allows up to 256 bit keys but can be taxing to a system. The results of using AES are highly dependent on a particular site and the amount of traffic traveling to and from that site.

Concerning traffic, CountyXYZ provides access to publicly available records and property data to whoever chooses to connect to our main web site. Citizens' sign up for and in some cases pay for login rights. The Login is used to access thousands of records that are stored in internal databases separate from our DMZ, which protects our web servers. Essentially a user is connecting to an external IP is redirected to an actual IP of a web server in our DMZ depending on the destination chosen. If the user is accessing public records then the authentication takes place in the DMZ and the user is redirected again to the requested internal database. Since our sites are known to receive thousands of hits a day one can see that this within itself can be taxing to a system. This gets to my point.

Using AES, I began to see a degradation of efficiency with the SEF. During peak times and when traffic through VPN was simultaneously taking place, data would pass through the tunnel much slower then expected. Further to my discovery, outside users complained of losing connection to public data. Also, I noticed that system performance would spike on the SEF and briefly hold near 100% of the system resources for processing power. These problems did not seem to occur when using DES or 3DES. However, since DES does not provide encryption considered to be secure and AES simply adds too much of a load on our enterprise firewall I had to make a choice to use 3DES.

IPSEC VPN tunnel creation:

Upon logging in to each device via your SRMC you will notice that Symantec Firewall/VPN products try to entice you with various "Wizards" one of them being the "S2S Tunnel Wizard". Do not fall into temptation. Using these wizards do provide some what of an ease for configuration but what they don't do is give you a clear idea of how all the pieces of the puzzle come together. However, "Wizards" can provide some ease of configuration for an inexperienced administrator. They can also be used as a tool to learn where everything is placed within the configuration after the wizard has been run. However, Symantec does not advise that inexperienced administrators alter the configuration of a firewall until properly trained. Case in point, access was

granted to an administrator at the first remote location. The administrator began fooling with the configuration via the “Wizard” without my knowledge. The administrator accessing the appliance was perfectly acceptable as long as notice was given so proper guidance could be available. The changes took place on the VelociRaptor appliance only. The administrator’s crafty use of the “Wizard” deemed the VelociRaptor non-functional and because the admin was not properly trained a proper explanation of the changes could not be given. This resulted in the need for a complete restore of the Sun/Linux OS version on the appliance as well as a restore from backup of the firewall configuration. After this event took place, all access other than mine was revoked on the remote firewalls. Anyway, any self respecting IT person would not be caught dead using a “Wizard” right? Hopefully I'm steering you in the right direction. Now with that said here is how it’s done.

Creating Entities and the Security Gateway:

First things first, you need to know who is going, how they will be traveling, and what route will be taken to get there. This leads us to the first step of our configuration. Here is where you will create internal nodes for your source to destination and the security gateways as tunnel endpoints. Creating entities is about the least complicated part of this process but before you start entering name and IP information you need to think about how to approach this. Entities can be entered as individuals, groups or subnets. Keep in mind that entities are nothing more than name, description and IP information on internal and external nodes. If the choice were that only select users would have access to the server, the best approach would be to create a group and then add individuals as needed. That way when rules are created you can apply the group to the rule rather than each user.

For the case of our organization, subnets were used in place of groups and individual entities. This option was selected because nearly every PC at the remote location would need access to internal services. Now to complete this step you will need to open your SRMC and connect the SEF. The next step is to expand out base components, right click “Network Entities” and create a new subnet for the remote location. An appropriate name will need to given so that it is easily associated with the remote location. Following the same step as above you will create individual entities and a group for your internal servers. Each entity will be added to the group and applied to a rule that I will give an example for later.

Now you will create the “Security Gateway” for endpoints of the tunnel. There is no surprise here. Follow the same steps as above but this time select new “Security Gateway”. The security gateway information will be the outside or WAN interface that connects the VPN tunnel. There are some differences that you will see in creating a security gateway over an entity. When you enter the IP address of the gateway you will notice an “Enable IKE” check box and IKE parameter settings. After entering the gateway IP address only check the “Enable IKE” option and leave the rest as default. IKE will be enabled due the choice of policy for this project.

Security policy:

The next step from here is to create a security policy. With the choice of Ipsec/IKE for and "Encapsulation protocol" out of the way we can get into setting up the policy. Symantec tries to help the user out when creating policies. Each device provides a series of default policies that can be used in production or as guidelines. With creating organizational policies, it would be best to create your own so you can modify as needed without mangling your defaults. Now from your SRMC you will expand "Virtual Private Networks", right click "VPN policies" and create a new "VPN policy". Name the policy so that it matches the scope of the project. Then select your "Encapsulation Protocol" which will be Ipsec/IKE. Rules will also be used for the tunnel so in order to have this added measure of control "Pass Traffic from the Secure Tunnel to the Proxy Services" must be checked. Once your EP is selected you will notice that 5 new tabs appear across the top of the open window. Select the Ipsec/IKE tab and enter your integrity and privacy preferences. The preferences for this project are SHA1 and 3DES as mentioned previously. Data compression is also an option from this screen. I have tested with none as well as LZS and the tunnel performed well in both cases. As stated in [RFC 1974] (8):

- The *LZS* algorithm is optimized to compress all file types as efficiently as possible. Even string matches as short as two octets are effectively compressed.

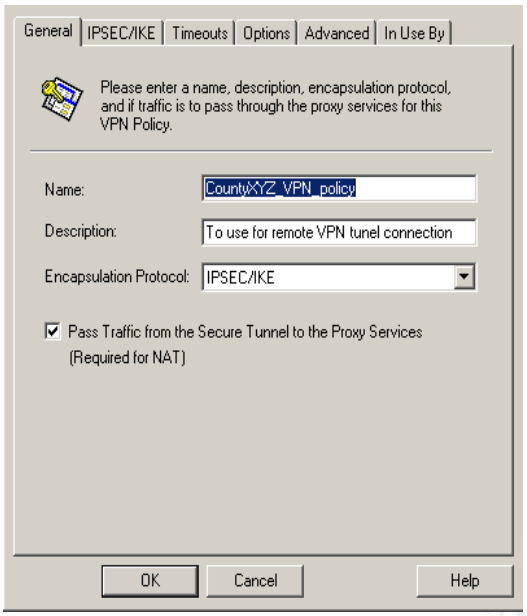
This compression technique will make the data more efficient when passing through the VPN tunnel. Electing to use this process will increase the load on system processors when in use. As stated in the Symantec Enterprise Firewall p.352 "LZS requires several CPU cycles to perform compression. For the type and amount of data that is transferred between the organizations as described by the paper, none was an acceptable choice. In other words no compression is used. Choosing compression or not would be something to experiment with within you own organization.

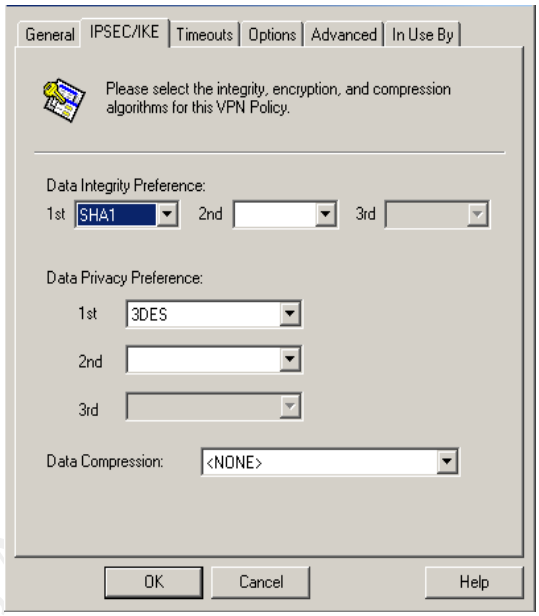
The next two tabs "Timeouts" and "Options" were left alone. They should only be adjusted if you are experiencing connectivity problems. Under the advanced tab, you will find the options of Encapsulation mode, Data Integrity and Perfect forward Secrecy or PFS. Selecting PFS will open up your choices for Diffie-Hellman Preference. "Diffie-Hellman is the standard IKE method of establishing shared keys. Group 1 and 2 are the Diffie-Hellman group numbers available for establishing these IKE session keys." (9) An easy way to look at these two choices is by to think of group 1 less secure less CPU power and group 2 being more secure and using more CPU power. More specifically there is a bit length difference of the shared keys between the two groups. Group 1 offers a length of 768 bits and group 2 offers a length of 1024 bits.

The tradeoff between these two choices would be the level of the security needed for generated keys as well a sacrifice of some processing power. In a project like this, one may want to experiment with each to determine which

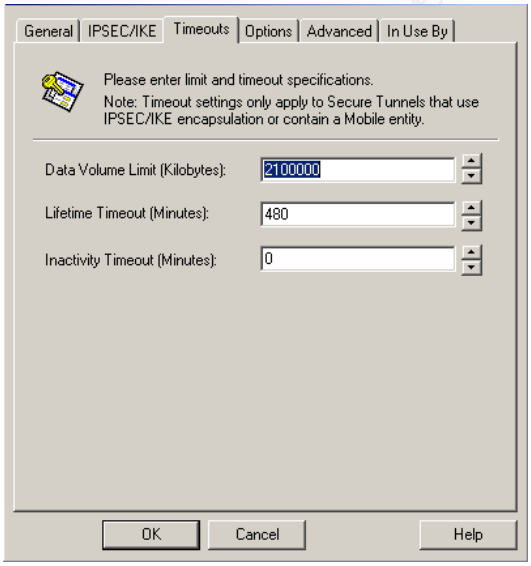
will be right for your organization and the sensitivity of the data that needs to be protected. Remember that once the policies are created they can be adjusted to the needs of each project. To continue with my organization's configuration, a series of screen shots is provided on the next page. Due to legibility issues, the sizes of the objects have to remain no smaller than they are now.

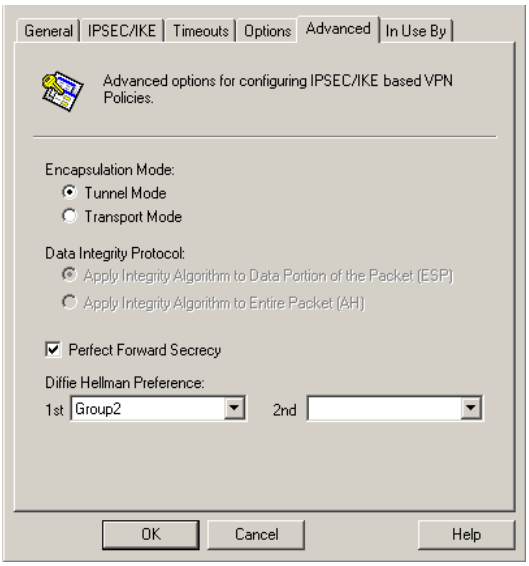
Figures 1 & 2





Figures 3 & 4

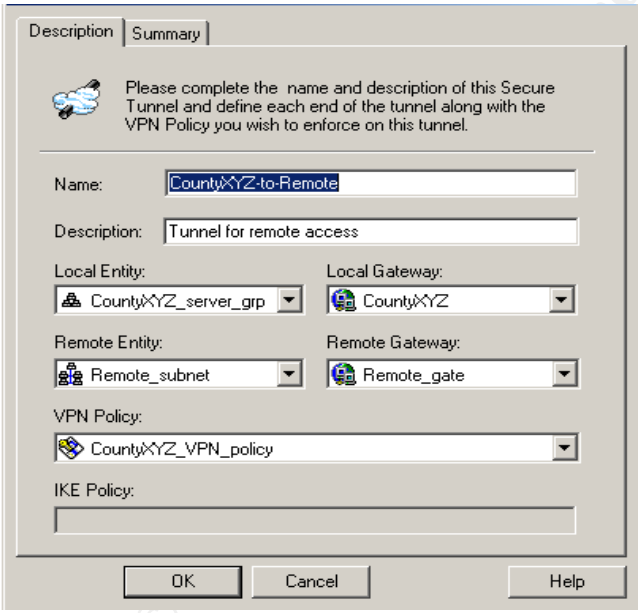




Creating the secure tunnel:

At this point, all local and remote entities, security gateways and policies should have been created and saved. That way when you create the tunnel all objects will be available for configuration. For VPN tunnel creation in short, you will select a previously configured local security gateway with your local entity, group or subnet. In reverse you will select the opposite for your remote gateway and subnet. Remember as previously mentioned that your Security Gateway is the outside interface of each device. To create the tunnel, which may begin to sound familiar, open your SRMC again and expand Virtual Private Networks. Next, right click Secure Tunnel and create a new Secure Tunnel. It's a good idea to keep with consistency when naming the tunnel. Make sure that the name is relevant to the project. You can also add a description for more information. Next select your local and remote entities and security gateways. Below these four selections you will then choose your previously created VPN policy. When using IKE, after you save this secure tunnel configuration, no further changes are needed to complete the tunnel on the local end. The screen shot shown below illustrates this task.

Figure 5

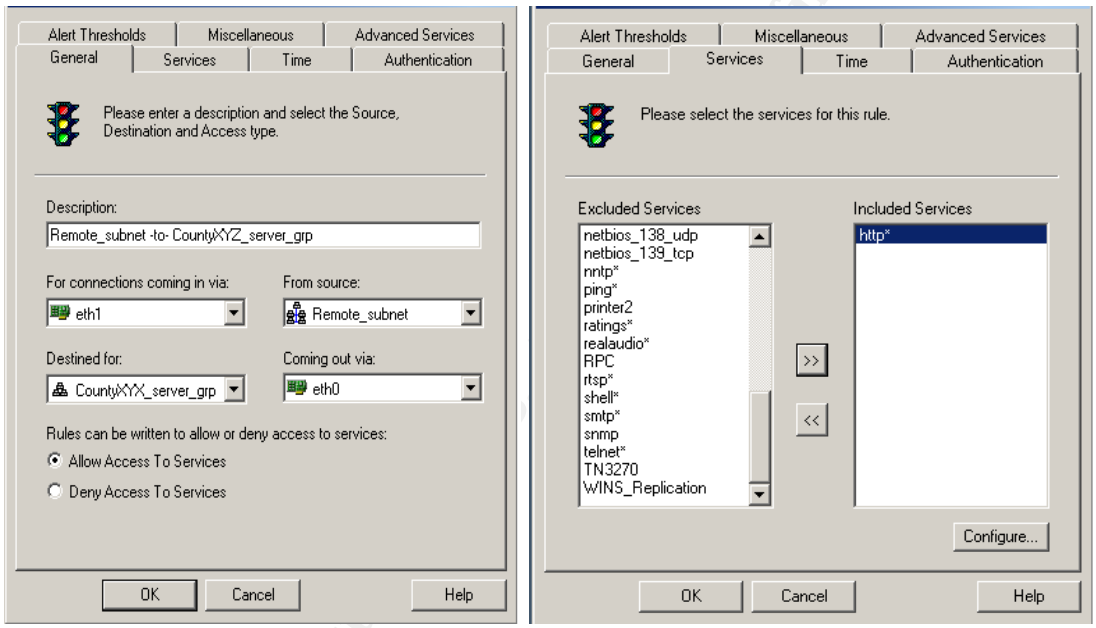


Creating rules:

To finish this process off we need to create rules that can be considered the gatekeeper to the tunnel. All the parts needed to create the rules should be in place if you have gotten this far. As we have been doing previously, connect via your SRMC and expand out access controls. Then you will select and right click rules to create a new rule. From here you will see a screen that

opens up to reveal a series of tabs. We will only be concerned with two of them. The first tab showing is the general tab. This is where most of the configuration takes place. Here is where a description, source to destination and interfaces will be added. When creating this rule from the local end of the tunnel, select the VPN tunnel and remote subnet for incoming connection. Then select the internal interface and internal group for the point for traffic passing through to the inside of the firewall. The next tab, services, is where you will add protocols to pass through the tunnel. As mentioned earlier, the internal servers within the local network have been set up for access via web interface so the only service currently configured for the CountyXYZ tunnel is http. As future need may arise, new rules can be added to satisfy requests or additional services can be added to existing rules. Observe the screen shots of the rule creation below.

Figures 6 & 7



If you have noticed, all examples up until now have been one sided. That's because you only need to understand one part of the equation in order to figure out the rest. The completion of this project only needed an opposite duplication of the procedures explained above on the remote Velociraptor. Once you are past this procedure, your secure Ipsec site-to-site VPN tunnel is complete. At this point you would have the remote users of the tunnel attempt a connection to your internal servers. In the case for CountyXYZ the users were presented with a prompt for username and password. This ensures that the VPN tunnel is working. The question now should be that if the data is encrypted or not.

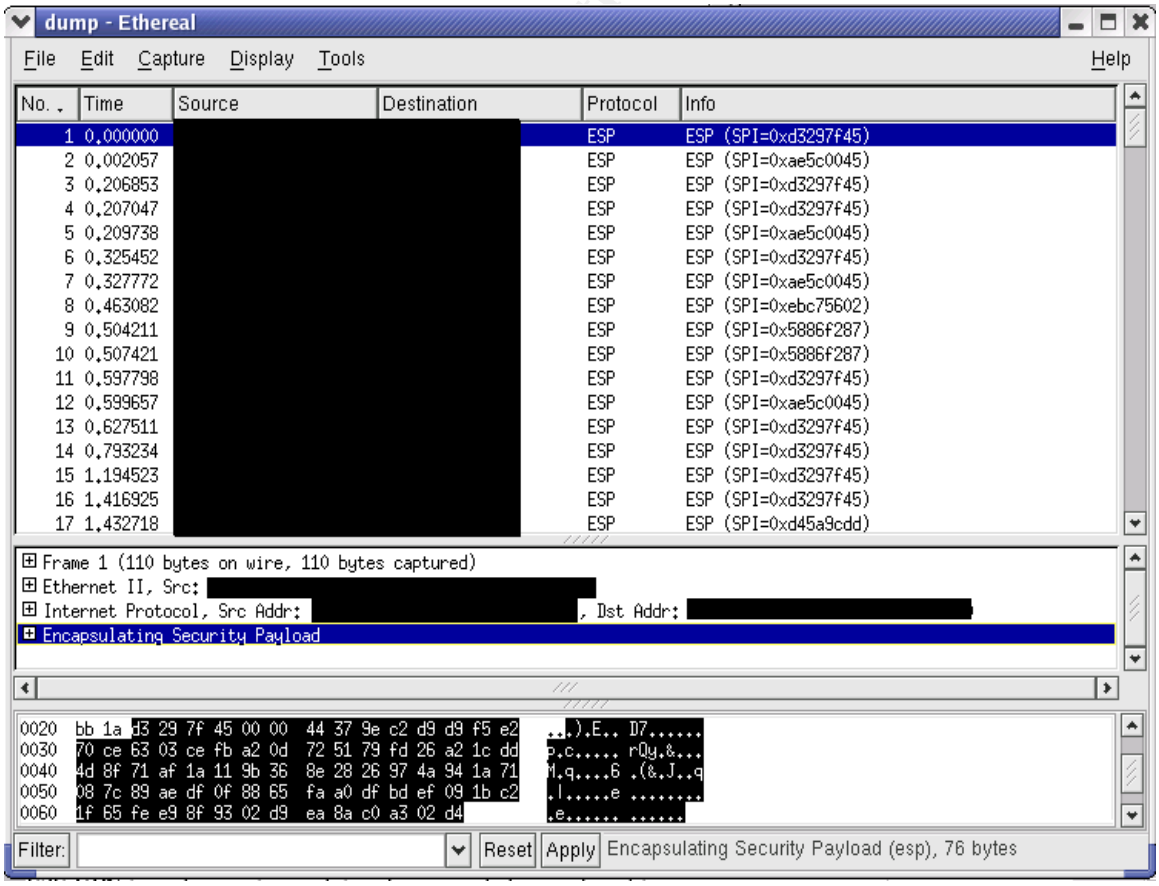
Verify that the data is encrypted:

Once connection is complete and you can access services, the next step should be to verify that the data is encrypted. There are numerous ways to complete this task. Two popular packet-capturing applications are TCPDUMP

and Ethereal. Which ever you choose will produce the same results. If you want to go the simple route, each of the Symantec devices come supplied with TCPDUMP which can be run and set up to watch any interface you like on each device. If run from a SEF, in our case, you would open a command prompt to start TCPDUMP. If run from a Velociraptor, you would use the built-in SRL, which is the UNIX equivalent of the command prompt.

In the case being presented in this paper, I used Ethereal running on a Redhat 8.0 laptop. Normally I use tcpdump running on each device for convenience but for this practical I chose Ethereal because of it's output to a GUI for presentation purposes. Now, to make sure I was gathering the correct data, I mirrored a port on a switch that was directly connected to the "outside" interface of the SEF. This would also be the same interface that the VPN tunnel passes through. From this mirrored port I connected my laptop and started up Ethereal. Then to gather data, the remote location attempts to access each of the internal servers and query records. The results are as expected and the data gathered reveals only the protocol ESP. This indicates that the data is in fact encrypted. Below you will see a small capture of data that proves that the data is encrypted. IP information has been removed for obvious reasons. The screen shot of the collected data is shown on the following page.

Figure 8



Wrap up:

With the completion of the site-to-site VPN tunnel project, many benefits came about as a result. One is that the use of a site-to-site VPN tunnel reduced the amount of man-hours that would have been needed to set-up client software on each of the municipality's machines. Plans for the first site alone were set for 20-30 users of the system with a possibility of adding more. This would have meant pulling someone away from a more important project in order to set up a new user. Multiply this number by the amount of potential users and you have a problem. Not to mention the client issues that can come up with the existing users. The creation of a VPN tunnel also provides simplified management that helps in making changes or troubleshooting a problem. For instance, say the grouping of our internal servers for remote access expands; access to these new internal servers can be managed by making a few simple global changes. Additionally, troubleshooting connectivity issues through the tunnel are narrowed to working on just two possible devices rather than from the client through any number of devices in between.

The site-to-site VPN tunnel has provided a secure way to transfer data between all locations participating in this project. This ensures that the sensitive data being access is not compromised from outside sources. The ability to use the Internet for this project has also increased speed and availability to view/update internal records. As a follow up, the remote users of the VPN connection have been pleased with its performance. Through close monitoring of each system and each of the system logs, no relevant errors are being recorded. Furthermore, due to the success of this project, access will be expanded to more internal nodes for authorized remote user access.

Conclusion:

Fortunately the timing was just right for me to complete this project. Configuration and Installation of the Velociraptor at the remote municipality location had been completed on an earlier date. This gave me an understanding of the Velociraptor configuration as well as the remote network layout. This should not be too much of an issue for someone completing a similar task. However, being involved in all of the steps will help your understanding of the project. Finally, Ipsec VPN is a good enhancement to your existing security policy when trying to expand the boundaries of your network.

References:

0 [IPSEC] Kent, S., and R. Atkinson, "Security Architecture for the internet protocol", RFC 2401 November 1998. http://rfc-2401.rfclist.com/rfc-2401.htm

1 [SHA-1] FIPS PUB 180-1, "Secure Hash Standard," National Institute of Standards and Technology, U.S. Department of Commerce, April 17, 1995. http://csrc.nist.gov/fips

2 [ESP] Kent, S., and R. Atkinson, "IP Encapsulating Security Payload", RFC 2406, November 1998. ftp://ftp.rfc-editor.org/in-notes/rfc2406.txt

3[AH] Kent, S., and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998. ftp://ftp.rfc-editor.org/in-notes/rfc2402.txt

4 [ICSA labs] http://www.icsalabs.com/html/communities/ipsec/lab/notes/1.0B/symantec_sevpn70.shtml

5 [3DES] National Institute of Standards and Technology, “Data Encryption Standard” FIPS PUB 46-3. October 25, 1999. http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

6 [IKE] Harkins, D., and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998. ftp://ftp.rfc-editor.org/in-notes/rfc2409.txt

7 [VPN] VPN Consortium “VPN Technologies: Definitions and Requirements”, January 2003. http://www.vpnc.org/vpn-technologies.pdf

8 [LZS] R. Friend and W. Simpson, “PPP Stac LZS Compression Protocol” RFC 1974, August 1996. http://rfc-1974.rfclist.com/rfc-1974.htm

9 Symantec Corporation, Symantec Enterprise Firewall, Symantec Enterprise VPN, configuration guide. Ehelp Corporation 2001. http://www.symantec.com

Additional Links:

http://www.tcpdump.org/

http://www.ethereal.com/