

Cisco Secure PIX 5.0 Firewall

IPSEC DESIGN AND IMPLEMENTATION GUIDE

Introduction

This is a major release for Cisco Secure PIX Firewall allowing PIX to be a complete End-to-End VPN solution. The PIX 5.0 now supports IPsec. The PIX implementation of IPsec (IP Security) is based upon the Cisco IOS IPsec implementation and provides compatibility with the Cisco IOS Router based IPsec. In addition, PIX also supports Windows 95, Windows 98, and NT 4.0 VPN users with the Cisco Secure VPN Client. The PIX IPsec supports both DES (56-bit) and 3DES (168-bit), with the addition of the appropriate encryption license key.

(56-bit) DES Encryption License: Is a \$0 item which can be selected and ordered to allow customers to enable DES encryption for their 5.0(1) software. All current shipping PIX models will support IPsec 56-bit DES with PIX 5.0 software: 515R,UR, and all 520. (See HW/SW compatibility in next section for older PIX models) PIX-VPN-DES.

Remember to register online for the IPsec Key you must have your Serial #!
<http://www.cisco.com/kobayashi/sw-center/internet/pix.shtml>

(168-bit) 3DES Encryption License: 3DES (168-bit) encryption is available for the additional price of \$1000. The part number for this software license key is PIX-VPN-3DES. (all current shipping models will be supported)

Client Software: The Cisco Secure VPN Client for DES or 3DES is available for purchase in three different packages: 100 user licenses for \$250, 1000 user licenses for \$1000, and unrestricted user licenses for \$2500. PIX Encryption license and client license bundles will also be available.

PIX firewall changes to note:

Privatelink commands are no longer supported. These commands are replaced with IPsec commands. **Note: if you have a PIX with the Privatelink card installed as a VPN you will need to manually convert the link configuration to an IPsec configuration.**

The Stateful Failover feature was enhanced so that the stateful connection information is passed on to the Standby unit during a failover. Stateful failover supports long-lived connections, which exclude http. **Note: an Ethernet dedicated port is required to allow stateful failover. The “failover link” command is used to specify which ethernet link to use.**

PIX Firewall now provides enhanced support for the H.323 RAS protocol, which supports Cisco's implementation of Voice over IP. **Note: This release of PIX does NOT support the Cisco skinny call manager protocol.**

PIX now supports a Firewall MIB that will be made available on CCO.

General

SOFTWARE AND HARDWARE REQUIREMENTS 4
 SOFTWARE AND HARDWARE PERFORMANCE NUMBERS 5
 IMPORTANT DESIGN POINTS AND CAVEATS 8
 DESIGN GUIDELINES 9

Sample Configurations

FIRST 3 EXAMPLES ARE PIX TO PIX IPSEC

1) SECURE TUNNEL BETWEEN TWO PIXS USING PRE-SHARED KEYS [NO NAT] 14
 2) SECURE TUNNEL BETWEEN TWO PIXS USING DIGITAL CERTIFICATES ISSUED BY VERISIGN [NO NAT] 18
 3) SECURE TUNNEL BETWEEN TWO PIXS USING DIGITAL CERTIFICATES ISSUED BY ENTRUST [NO NAT] 22

NEXT 3 EXAMPLES ARE REMOTE USERS WITH CISCO SECURE VPN CLIENT CONNECTING TO PIX

4) SECURE TUNNEL BETWEEN AN IPSEC CLIENT AND A PIX USING PRE-SHARED KEYS AND VIRTUAL IP ADDRESSES MANUALLY ASSIGNED] (NO CERTIFICATE REQUIRED) 27
 5) SECURE TUNNEL BETWEEN AN IPSEC CLIENT AND A PIX USING PRE-SHARED KEYS, MODE CONFIG AND AAA FUNCTIONS (NO CERTIFICATE REQUIRED) 31
 6) SECURE TUNNEL BETWEEN AN IPSEC CLIENT AND A PIX USING VERISIGN DIGITAL CERTIFICATES, MODE CONFIG AND AAA FUNCTIONS 38
 CRYPTO VOCABULARY 44
 UNDERSTANDING MODE CONFIG AND ASSIGNING INTERNAL IP ADDRESSES TO CLIENTS 50
 PIX CRYPTO COMMAND REFERENCE 54

Hardware/Software Requirements

PIX version 5.0 software

Hardware Requirements

2MB Flash and 32MB of RAM

Encryption IPsec VPN with PIX:

An encryption card is NOT required to run IPsec on PIX. The 5.0 is capable of running DES and 3DES in software.

If a customer has a PL2 card the 56DES drivers will take advantage of that card and you can expect 56DES to run significantly faster. Please be aware the PL2 card will provide the most benefit for 56DES, 3DES will get a performance increase with the new encryption card in Q4.

Hardware/Software compatibility to run PIX 5.0:

Please check www.cisco.com to review the H/W-S/W compatibility matrix for PIX. Use the chart below as a guideline only:

Please note PIX has a 128RAM upgrade and it comes in two versions for the different model series of the PIX. Please provide your Serial # to order the correct 128RAM upgrade.

If you have	You need to
PIX Classic and all older models below Ser #0600400	These units cannot be upgraded and they need to be RMA
PIX Classic Ser # 06004001 through 06009399	<u>You need to upgrade with 128RAM</u> PIX-MEM-UPG-128=
PIX 10000 Ser # 18000000 through 18004999	<u>You need to upgrade with 128RAM</u> PIX-MEM-UPG-128=
PIX 510 Ser # 16009400 and higher	<u>You need to upgrade with 128RAM</u> PIX-MEM-5XX-128=
PIX 520 Ser # 1800500 through 18013334 comes with 32MB RAM	No new memory required you need only upgrade to 5.0
PIX 520 Rev B0 and higher comes with 32MB RAM	No new memory required you need only upgrade to 5.0
PIX 520 Rev C0 and higher comes with 128MB RAM	No new memory required you need only upgrade to 5.0
PIX 515 R & UR	No new memory needed you need only upgrade to 5.0

Also check http://www-tac.cisco.com/Support_Library/Hardware/PIX/Troubleshooting/PIX_serenum.htm

Performance, Memory and CPU Considerations

Here is the PIX Firewall 5.0.IPSec Throughput Performance Test Results for PIX-515 (200MHz) and PIX-520 (350MHz)

We used the following topology and hardware to find out the throughput.

Client1-9-----PIX-515 200MHz/64MB-----PIX-515 200MHz/64MB-----server

Client1-9-----PIX-520 350MHz/32MB-----PIX-520 350MHz/32MB-----server

We tested both UDP and TCP traffic with varying packet sizes 300 bytes and ~1400 bytes. Traffic travels only one way, and there is no re-keying while transferring files.

We found that throughput, with and without PL2 does not make any difference with 3DES but it does make a difference with DES protocol. This is because 3DES encryption is performed only in software whereas DES is achieved in the PL2 card only if the card is present.

There is no difference in throughput for 300 and 1400 bytes TCP packets. Therefore, the test result of the 300 bytes TCP packets is omitted from the result table. But there is a big difference in throughput for UDP traffic as the packet size varies.

Note: All traffic is IPSec traffic. There is no traffic clearly with firewall filters to process the traffic. Take for example ESP-DES-SHA (with PL2 card) which shows 49.8 MB on a PIX 520 for TCP traffic. A file copied from client to server with 56Bit DES and PL2 card ran at 49.9MB

	Pix-515 (200Mhz)			Pix-520 (350Mhz)		
	Tcp 1400 bytes	Udp 1400 bytes	Udp 300 bytes	Tcp 1400 bytes	Udp 1400 bytes	Udp 300 bytes
Ah-md5	50.0	73.2	30.6	98.8	98.8	84.4
Ah-sha	33.2	46.6	20.6	95.5	98.8	63.1
Esp-des	21.5	25.4	19.3	50.7	53.7	43.3
Esp-des (PL2)	53.0	68.7	30.7	77.3	89.6	50.4
Esp-des-md5	18.0	21.5	15.3	43.0	46.5	39.5
Esp-des-md5 (PL2)	34.6	46.3	21.5	59.3	70.1	41.1
Esp-des-sha	15.3	18.3	12.5	37.3	40.2	35.4
Esp-des-sha (PL2)	26.5	34.6	16.5	49.8	59.1	35.7
Esp-3des	11.1	12.2	11.3	22.2	22.9	23.3
Esp-3des-md5	10.0	11.3	9.7	20.8	21.4	22.6
Esp-3des-sha	9.1	10.5	8.6	19.3	20.1	19.8

Notes On Performance, CPU, Memory and SAs:

1. IPSec processing such as encryption, decryption, hashing etc is required at each packet.
2. Encrypted packets will probably be authenticated, which means that there are two cryptographic operations being performed for every packet.
3. Diffie-Hellman is used only by IKE and only during the initial tunnel setup for IKE. IPSec does not use Diffie-Hellman. Authentication algorithms for IPSec (SHA-1 and MD5) are relatively fast. The slowest IPSec algorithm is Triple-DES.

In addition, the Diffie-Hellman key exchange used in IKE is an exponentiation of very large numbers (between 768 and 1024 bytes) and can take up to several seconds. Performance of the RSA is dependent on the size of the prime number chosen for the RSA key pair.

As a general rule any PIX Model today can support over 2000 IPSec SAs. This number will increase in the coming months as new encryption hardware is added, memory, and CPU. For example: the all new PIX 520 ships with 128RAM and the new PIX 515 ships with 64RAM, older models have much less memory but can run IPSec.

Important Design Points and Caveats

General

- **IPSec tunnels are terminated on the outside interface.** This means that the remote identity networks are the global addresses when NAT is used. In the next release of IPSec for PIX you will be able to terminate IPSec traffic on ANY interface. The outside interface restriction should be noted in all designs with this release. IOS IPSec and PIX IPSec will become streamlined in the PIX 5.1 release later this year. Today you will notice nominal CLI differences.
- **IPSec terminate on the inside is available in 5.0.2.** If you have a customer that wants free inside network access this is the release you should use. An example would be NT Network neighborhood access. Please inform the customer that PIX in this configuration is no longer inspecting IPSec traffic as a firewall. PIX is authorizing it inside and only using IKE to authenticate. Both options should be fully explained; terminate both on the inside and outside.
- **Remember to register online for your free 56bit Key or \$1000.00 168bit key** <http://www.cisco.com/kobayashi/sw-center/internet/pix.shtml>
- You will not be able to do any VPN configuration options until you get your key for 5.0 IPSec. Reboot your PIX and enter new key.
- IPSec
- **Access-lists are required to trigger IPSec.** The access list defines the traffic to protect in an outbound direction. You can also use access-list to filter per-interface traffic via the access-group command. **In version 5.0, protocol and ports numbers are not supported in access lists.** You can specify host to host; network to network, network to host, network to any, any to network, or any to any.
- **The Static command is required to provide NAT function for IPSec.** You will need static to enable the remote host to connect to the internal host from the outside. However, if the connection will be initiated from the inside, then NAT and global will suffice. Static is not required if you are terminating on the inside.
- **The Conduit command is NOT required for IPSec traffic.** It is still used for the firewall function. This has been changed to maintain IOS compatibility. IPSec traffic is not trusted and will be denied unless explicit conduits are configured for the “decapsulated” IPSec packet. See the Sysopt connection permit-ipsec below to implement a trust policy that will not require conduits.
- **Sysopt connection permit-IPSec:** This command is important in all IPSec configurations. This command tells PIX to Implicitly trust IPSec traffic and bypass the checking of an associated conduit for IPSec connections. Outbound is not applicable in this release. If **sysopt connection permit-ipsec** is not configured, you must explicitly configure the conduit to enable IPSec traffic to traverse the PIX Firewall. This method would allow port restrictions. Please note the design implications of both methods
- **IP local pool:** This is an important command for assigning a pool of IP addresses to be used as your Virtual IP addresses for your VPN users (mode config).

Important Design Points and Caveats continued:

IKE

- RSA-Encryption or commonly known as encrypted nonces are not supported on PIX. The default policy and the default values for configured policies do not show up in the configuration when you issue a write terminal command. Instead, to see the default policy and any default values within configured policies, use the show isakmp policy command.)

CA

- When using RSA Signatures [CA], make sure to invoke *ca save all*, otherwise ca related information will be deleted on a reboot. It is important to note wr mem is not saving CA data. **Remember to always ca save all and wr mem.**
- **Domain-name** command is only required in CA configurations. Like routers, only one root CA can be installed at a given time.

Mode Config

- Only one pool can be defined at this time.

Telnet ip address [netmask] [if name]

- This is important. We can now specify outside interface for telnet, and enable IPsec. This allows for remote management to be encrypted by using the Cisco Secure VPN Client. This does not solve inside Telnet as it is still not encrypted, in 5.1 you will be able to encrypt Telnet on the inside to PIX. At a minimum, the crypto map command must be configured to specify an interface name with the telnet command.

PIX Firewall Manager:

- The PIX firewall Manager will be supported in this release, however no IPsec commands will be supported. Thus you can continue to use PFM for Firewall functions. In the 2.0 Release of Cisco Security Manager, CSM will support IPsec commands and PIX firewall commands in one GUI. CSM will require PIX 5.1 code to work. PFM version 4.3(2)c now works with PIX Firewall version 5.0, but does not support any new feature or command from either version 4.4 or version 5.0.

Trouble shooting:**show ca mypubkey rsa**

Displays own RSA public key

show ca identity

Displays CA identity information

show ca certificate

Displays CA certificates.

show isakmp policy

Displays IKE policies.

show isakmp sa

Displays current IKE sa's.

show ipsec sa

Displays ipsec sa's.

show crypto map

Displays map

show crypto dynamic-map

Displays map dynamic-map

debug crypto isakmp**debug crypto ipsec****debug crypto ca**

Design Guidelines

1. If the remote peer is not PIX and does not support IKE,

Then use manual IPsec. In this case IKE is disabled. If you disable IKE, you will have to make these concessions at the peers:

You must manually specify all the IPsec security associations in the crypto maps at all peers. (Crypto map configuration is described in the "Configuring IPsec Network Security" chapter.) The peers' IPsec security associations will never time out for a given IPsec session. During IPsec sessions between the peers, the encryption keys will never change. Anti-replay services will not be available between the peers. Certification Authority (CA) supports cannot be used.

2. How Do IKE Peers agree Upon a Matching Policy?

When the IKE negotiation begins, IKE looks for an IKE policy that is the same on both peers. The peer that initiates the negotiation will send all its policies to the remote peer and the remote peer will try to find a match. The remote peer looks for a match by comparing its own highest priority policy against the other peer's received policies. The remote peer checks each of its policies in order of its priority (highest priority first) until a match is found.

A match is made when both policies from the two peers contain the same encryption, hash, authentication, and Diffie-Hellman parameter values, and when the remote peer's policy specifies a lifetime less than or equal to the lifetime in the policy being compared. (If the lifetimes are not identical, then the shorter lifetime---from the remote peer's policy---will be used.)

3. Which value should you select for each IKE Parameter?

You can select certain values for each parameter, per the IKE standard. But why choose one value over another?

If you are inter-operating with a device that supports only one of the values for a parameter, your choice is limited to the other device's supported value. Aside from this, there is often a trade-off between security and performance, and many of these parameter values represent such a trade-off. You should evaluate the level of your network's security risks and your tolerance for these risks. Then the following tips might help you select which value to specify for each parameter.

The encryption algorithm currently has two options: 56-bit DES-CBC and 168-bit 3DES. 3DES is much secure but has export restrictions and performance is slower than 56-bit DES.

The hash algorithm has two options: SHA-1 and MD5. MD5 has a smaller digest (128 bit) and is considered to be slightly faster than SHA-1 (160-bit).

The authentication method has two options: RSA signatures and pre-shared keys. RSA signatures provide non-repudiation for the IKE negotiation. Also RSA signatures requires use of a Certification Authority (CA). Using a CA can dramatically improve the manageability and scalability of your IPsec network. Pre-shared keys are clumsy to use if your secured network is large, and do not scale well with a growing network. However, they do not require use of a Certification Authority, as do RSA signatures, and might be easier to set up in a small network with fewer than 10 nodes. If you are using a wild-card pre-shared key, make sure to attach it with user authentication.

The Diffie-Hellman group identifier has two options: 768-bit or 1024-bit Diffie-Hellman. 1024-bit Diffie-Hellman is harder to crack, but requires more CPU time to execute.

SA Lifetimes: The range for the PIX is 60 sec - 86400 seconds for ISAKMP SAs. As a general rule, the shorter the lifetime (up to a point), the more secure your IKE negotiations will be. However, with longer lifetimes, future IPsec security associations can be set up more quickly, reducing processor overhead.

4. Number of IKE policies

You can create multiple IKE policies on the PIX, each with a different combination of parameter values. This enables PIX to accept IKE connections from multiple peers. For each policy that you create, you assign a unique priority (1 through 10,000, with 1 being the highest priority). If you do not specify a value for a parameter, the default value is assigned.

5. ISAKMP Identity especially for IRE Client and RSA Signatures

When two peers use IKE to establish IPsec security associations, each peer sends its identity to the remote peer. Each peer sends either its fully qualified domain name or its IP address, depending on how you set the router's ISAKMP identity.

By default, a peer's ISAKMP identity is the peer's IP address. If appropriate, you could change the identity to be the peer's host name instead. As a general rule, set all peers' identities the same way---either all peers should use their IP address, or all peers should use their host name. If some peers use their host name and some peers use their IP address to identify themselves to each other, IKE negotiations could fail if a remote peer's identity is not recognized and a DNS lookup is unable to resolve the identity.

When using digital certificates, specify **domain name** as the ID type when specifying the remote gateway and specify the ip address when using pre-shared keys.

To set a peer's ISAKMP identity, use the following commands in global configuration mode:

Step	Command	Purpose
1.	crypto isakmp identity {address hostname}	At the local peer: Specify the peer's ISAKMP identity by IP address or by hostname.

6. IPsec lifetimes

IPsec security associations use shared secret keys. These keys and their security associations time out together.

Assuming that the particular crypto map entry does not have lifetime values configured, when the PIX requests new security associations during security association negotiation, it will specify its global lifetime value in the request to the peer;

It will use this value as the lifetime of the new security associations. When the PIX receives a negotiation request from the peer, it will use the smaller of the lifetime values proposed by the peer or the locally configured lifetime value as the lifetime

of the new security associations.

There are two lifetimes: a "timed" lifetime and a "traffic-volume" lifetime. The security association expires after the first of these lifetimes is reached.

If you change a global lifetime, the change is only applied when the crypto map entry does not have a lifetime value specified. The change will not be applied to the existing security associations, but will be used in subsequent negotiations to establish new security associations. If you want the new settings to take effect sooner, you can clear all or part of the security association database by using the clear crypto sa command.

To change the global timed lifetime, use the “crypto ipsec security-association lifetime seconds” form of the command. The timed lifetime causes the security association to time out after the specified number of seconds has passed.

To change the global traffic-volume lifetime, use the “crypto ipsec security-association lifetime kilobytes” form of the command. The traffic-volume lifetime causes the security association to time out after the specified amount of traffic (in kilobytes) has been protected by the security associations' key.

Shorter lifetimes can make it harder to mount a successful key recovery attack, since the attacker has less data encrypted under the same key to work with. However, shorter lifetimes require more CPU processing time for establishing new security associations.

The lifetime values are ignored for manually established security associations (security associations installed using an IPSec-manual crypto map entry).

How These Lifetimes Work

The security association (and corresponding keys) will expire according to whichever occurs sooner, either after the number of seconds has passed (specified by the seconds keyword) or after the amount of traffic in kilobytes has passed (specified by the kilobytes keyword).

A new security association is negotiated before the lifetime threshold of the existing security association is reached, to ensure that a new security association is ready for use when the old one expires. The new security association is negotiated either 30 seconds before the seconds lifetime expires or when the volume of traffic through the tunnel reaches 256 kilobytes less than the kilobytes lifetime (whichever occurs first).

If no traffic has passed through the tunnel during the entire life of the security association, a new security association is not negotiated when the lifetime expires. Instead, a new security association will be negotiated only when IPSec sees another packet that should be protected.

The example below shortens both lifetimes, because the administrator feels there is a higher risk that the keys could be compromised. The timed lifetime is shortened to 2,700 seconds (45 minutes), and the traffic-volume lifetime is shortened to 2,304,000 kilobytes (10 megabytes per second for one half hour).

```
crypto IPSec security-association lifetime seconds 2700
crypto IPSec security-association lifetime kilobytes 2304000
```

7. When to configure a crypto map set.

IPSec crypto maps link together the following:

- What traffic should be protected
- Which IPSec peer(s) the protected traffic can be forwarded
- Which transform sets are acceptable for use with the protected traffic
- How keys and security associations should be used/managed

Multiple crypto maps entries with the same map-name form a crypto map set:

A crypto map set is a collection of crypto map entries each with a different seq-num but the same map-name. Therefore you could have certain traffic forwarded to one IPSec peer with specified security applied to that traffic, and other traffic forwarded to the same or a different IPSec peer with different IPSec security applied. To accomplish this you would create two crypto maps, each with the same map-name, but each with a different seq-num.

The number you assign to the seq-num argument should not be arbitrary. This number is used to rank multiple crypto map entries within a crypto map set. Within a crypto map set, a crypto map entry with a lower seq-num is evaluated before a map entry with a higher seq-num; that is, the map entry with the lower number has a higher priority.

8. Use of Dynamic Crypto Maps

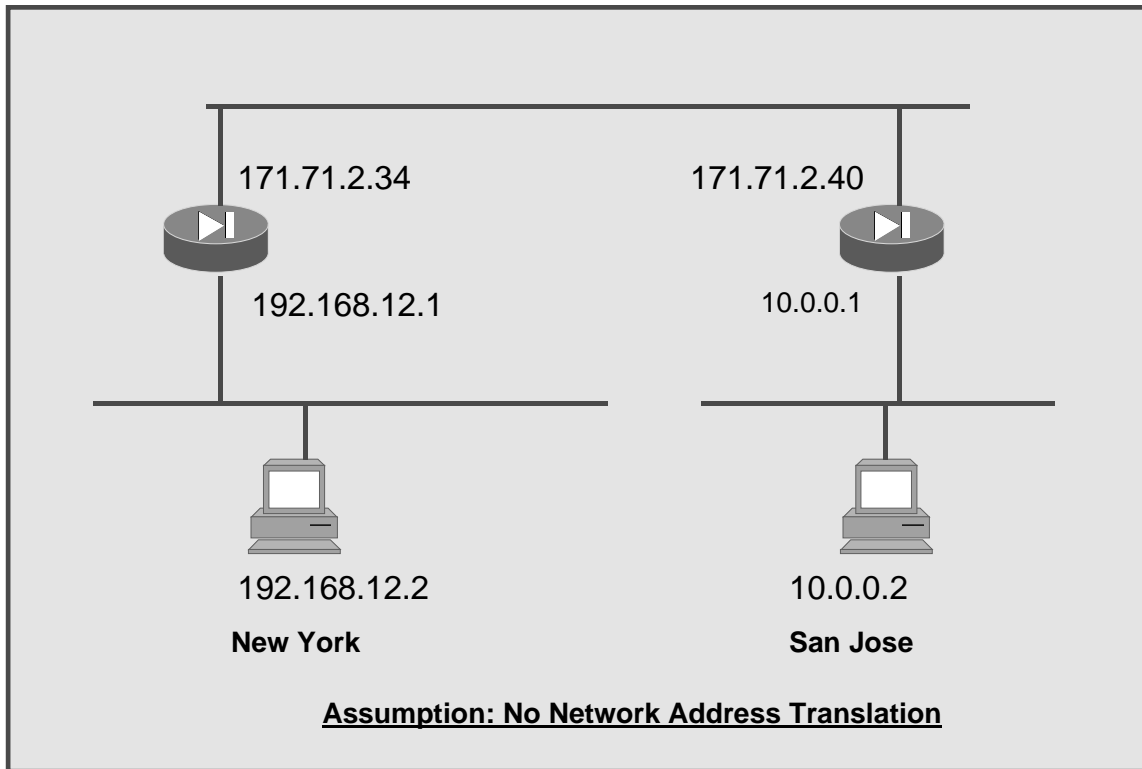
Use dynamic crypto maps when you do not know the IP address of the peer. If you want to allow travelling clients access to your network, you should use dynamic crypto maps.

You should make crypto map entries which reference dynamic map sets the lowest priority map entries, so that inbound security association negotiations requests will try to match the static maps first. Only after the request does not match any of the static maps do you want it to be evaluated against the dynamic map set.

Sample Configurations

Examples 1 –3 are based on the following topology. These examples will assume all traffic destined for New York from San Jose and all traffic from San Jose destined for New York is to be 3DES (168bit) encrypted.

Please note that in the next 3 examples if you are running the PIX 5.0.2 Release you would have a choice to avoid the network static commands and terminate IPsec on the inside. Both methods are allowed:



Example #1: PIX to PIX VPN Tunnel using Pre-Shared Keys: No Network Address Translation

➤ Enter the following crypto commands on the San Jose PIX

1. Define the hostname

```
hostname SanJose
```

2. Define the Domain Name (optional only required in CA configurations)

```
domain-name sisu.cisco.com
```

3. Create a Net Static

```
static (inside,outside) 10.0.0.0 10.0.0.0
```

4. Configure ISAKMP policy

```
isakmp enable outside  
isakmp policy 8 authentication pre-share  
isakmp policy 8 encr 3des
```

5. Configure pre-shared Key and associate with the peer

```
crypto isakmp key cisco1234 address 171.71.2.34
```

6. Configure IPSec supported transforms

```
crypto ipsec transform-set strong esp-3des esp-sha-hmac
```

7. Create an access list

```
access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0  
                        (source)           (destination)
```

8. Define a crypto map

```
crypto map newyork 10 IPSec-isakmp  
crypto map newyork 10 match address 80  
crypto map newyork 10 set transform-set strong  
crypto map newyork 10 set peer 171.71.2.34
```

9. Apply the crypto map to the interface

```
crypto map newyork interface outside
```

10. Tell PIX to Implicitly trust IPSec traffic

```
sysopt connection permit-IPSec
```

➤ Snapshot of PIX configuration on San Jose

```

PIX Version 5.0(0)204
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 10.0.0.0 10.0.0.0 netmask 255.0.0.0 0 0
access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map newyork 10 IPsec-isakmp
crypto map newyork 10 match address 80
crypto map newyork 10 set peer 171.71.2.34
crypto map newyork 10 set transform-set strong
crypto map newyork interface outside
isakmp enable outside
isakmp key cisco1234 address 171.71.2.34 netmask 255.255.255.255
isakmp policy 8 authentication pre-share
isakmp policy 8 encryption 3des
telnet timeout 5
terminal width 80
sysopt connection permit-IPsec
Cryptochecksum:e60634cd3f122e741acaald4e1c41bcf
: end

```


Configuration snapshot on PIX Firewall [New York]

```
PIX Version 5.0(0)204
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjYt7RRXU24 encrypted
passwd 2KFQnbNIdL.2KYOU encrypted
hostname NewYork
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
ip address outside 171.71.2.34 255.255.255.224
ip address inside 192.168.12.1 255.255.255.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 192.168.12.0 192.168.12.0 netmask 255.255.255.0 0 0
access-list 90 permit ip 192.168.12.0 255.255.255.0 10.0.0.0 255.0.0.0
no rip outside passive
no rip outside default
rip inside passive
no rip inside default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
no floodguard enable
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map toSanJose 20 IPsec-isakmp
crypto map toSanJose 20 match address 90
crypto map toSanJose 20 set peer 171.71.2.40
crypto map toSanJose 20 set transform-set strong
crypto map toSanJose interface outside
isakmp enable outside
isakmp key cisco1234 address 171.71.2.40 netmask 255.255.255.255
isakmp policy 9 authentication pre-share
isakmp policy 9 encryption 3des
telnet timeout 5
terminal width 80
sysopt connection permit-IPsec
Cryptochecksum:3c2f9a7f8705c4ac721cdd5d9e2bb643
: end
[OK]
NewYork#
```

Example #2: PIX to PIX VPN Tunnel using Verisign Digital Certificates

➤ Configuration on PIX San Jose

1. Define the hostname

hostname SanJose

2. Define the Domain Name (required in CA configurations)

domain-name sisu.cisco.com

3. Create a Net Static

static (inside,outside) 10.0.0.0 10.0.0.0

4. Configure ISAKMP policy

isakmp enable outside
isakmp policy 8 auth rsa-signature

5. Define Verisign related enrollment commands.

The IP address of onsiteIPSec.verisign.com server is 205.139.94.230

ca identity sisu.cisco.com 205.139.94.230

ca configure sisu.cisco.com ca 1 20 crloptional

[where 1 is retry period and 20 is retry count. crl optional says no crl checking]

6. Generate RSA key pair

ca generate rsa key 1024

7. Get the public key and the certificate of the CA server

ca authenticate sisu.cisco.com

8. Contact your CA Administrator and send the your certificate request.

ca enroll sisu.cisco.com cisco

[where cisco is a challenge password. Can be anything]

9. Configure IPSec supported transforms

crypto IPSec transform-set strong esp-3des esp-sha-hmac

10. Save Keys, certificates and CRL's in flash. Very Important

ca save all

11. Create a partial access list

access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0

(source)

(destination)

12. Define a crypto map

crypto map newyork 10 IPSec-isakmp
crypto map newyork 10 match address 80
crypto map newyork 10 set transform-set strong
crypto map newyork 10 set peer 171.71.2.34

13. Apply the crypto map to the interface crypto map toNewYork interface outside

14. Tell PIX to Implicitly trust IPSec traffic sysopt connection permit-IPSec



➤ Snapshot of PIX configuration on San Jose

```

PIX Version 5.0(0)204
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjLyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 10.0.0.0 10.0.0.0 netmask 255.0.0.0 0 0
access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map newyork 10 IPsec-isakmp
crypto map newyork 10 match address 80
crypto map newyork 10 set peer 171.71.2.34
crypto map newyork 10 set transform-set strong
crypto map newyork interface outside
isakmp enable outside
isakmp key cisco1234 address 171.71.2.34 netmask 255.255.255.255
isakmp policy 8 encryption 3des
ca identity sisu.cisco.com 205.139.94.230:cgi-bin/pkiclient.exe
ca configure sisu.cisco.com ca 1 20 crloptional
sysopt connection permit-IPSec
telnet timeout 5
terminal width 80
Cryptochecksum:e60634cd3f122e741acaa1d4e1c41bcf
: end

```


Sample Configuration on New York

```

PIX Version 5.0(0)204
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname NewYork
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
ip address outside 171.71.2.34 255.255.255.224
ip address inside 192.168.12.1 255.255.255.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 192.168.12.0 192.168.12.0 netmask 255.255.255.0 0 0
access-list 90 permit ip 192.168.12.0 255.255.255.0 10.0.0.0 255.0.0.0
no rip outside passive
no rip outside default
rip inside passive
no rip inside default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
no floodguard enable
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map toSanJose 20 IPsec-isakmp
crypto map toSanJose 20 match address 90
crypto map toSanJose 20 set peer 171.71.2.40
crypto map toSanJose 20 set transform-set strong
crypto map toSanJose interface outside
isakmp enable outside
isakmp policy 9 encryption 3des
ca identity sisu.cisco.com 205.139.94.230:cgi-bin/pkiclient.exe
ca configure sisu.cisco.com ca 1 100 crloptional
sysopt connection permit-IPsec
telnet timeout 5
terminal width 80
Cryptochecksum:113f7b32e7a652bfd9b4175a594a7f19
: end
[OK]

```

Example #3: PIX to PIX VPN Tunnel using Entrust Digital Certificates

➤ Configuration on PIX San Jose

1. Define the hostname

```
hostname SanJose
```

2. Define the Domain Name (required in CA configurations)

```
domain-name sisu.cisco.com
```

3. Create a Net Static

```
static (inside,outside) 10.0.0.0 10.0.0.0
```

4. Configure ISAKMP policy

```
isakmp enable outside
isakmp policy 8 auth rsa-signature
```

5. Define Entrust Server related enrollment commands.

The IP address of the entrust server is 192.150.50.132. Second address is for ldap query server.

```
ca identity my_nickname 192.150.50.132 192.150.50.132
```

```
ca configure my_nickname ra 1 20 crloptional
```

[where 1 is retry period and 20 is retry count. crl optional says no crl checking]

6. Generate RSA key pair

```
ca generate rsa specialkey 512
```

7. Get the public key and the certificate of the CA server

```
ca authenticate abcd
```

8. Contact your CA Administrator and send the your certificate request.

```
ca enroll abcd cisco
```

[where cisco is a challenge password. Can be anything]

9. Configure IPsec supported transforms

```
crypto IPsec transform-set strong esp-3des esp-sha-hmac
```

10. Save Keys, certificates and CRL's in flash. Very Important

```
ca save all
```

11. Create an access list

```
access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0
                        (source)          (destination)
```

12. Define a crypto map

```
crypto map newyork 20 IPsec-isakmp
crypto map newyork 20 match address 80
crypto map newyork 20 set transform-set strong
crypto map newyork 20 set peer 171.71.2.34
```

13. Apply the crypto map to the interface

```
crypto map newyork interface outside
```

14. Tell PIX to implicitly trust IPsec traffic

sysopt connection permit-IPSec

➤ Sample Configuration on San Jose

```

PIX Version 5.0(0)204
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
interface ethernet3 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 10.0.0.0 10.0.0.0 netmask 255.0.0.0 0 0
access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map newyork 10 IPsec-isakmp
crypto map newyork 10 match address 80
crypto map newyork 10 set peer 171.71.2.34
crypto map newyork 10 set transform-set strong
crypto map newyork interface outside
isakmp enable outside
isakmp policy 8 authentication pre-share
isakmp policy 8 encryption 3des
ca identity abcd 192.150.50.132:cgi-bin/pkiclient.exe 192.150.50.132
ca configure abcd ra 1 100 crloptional
sysopt connection permit-IPSec
telnet timeout 5
terminal width 80
Cryptochecksum:e720716d5760526739c043980f3b7546

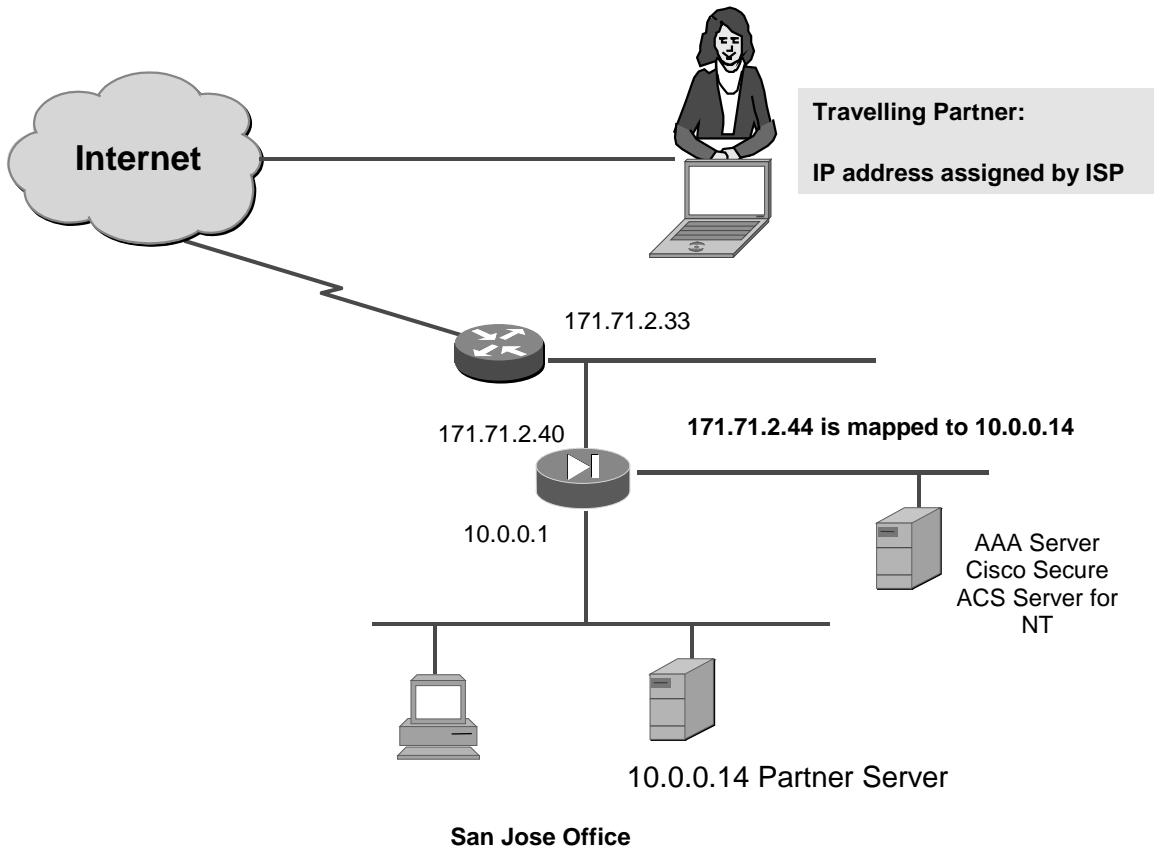
```

➤ PIX to PIX VPN Tunnel using Entrust Digital Certificates continued

Snapshot of PIX configuration on NewYork

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname NewYork
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging on
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
ip address outside 171.71.2.34 255.255.255.224
ip address inside 192.168.12.1 255.255.255.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
nat (inside) 0 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 192.168.12.0 192.168.12.0 netmask 255.255.255.0 0 0
access-list 90 permit ip 192.168.12.0 255.255.255.0 10.0.0.0 255.0.0.0
no rip outside passive
no rip outside default
rip inside passive
no rip inside default
route outside 10.0.0.0 255.0.0.0 171.71.2.40 1
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
no floodguard enable
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto map toSanJose 20 IPsec-isakmp
crypto map toSanJose 20 match address 90
crypto map toSanJose 20 set peer 171.71.2.40
crypto map toSanJose 20 set transform-set strong
crypto map toSanJose interface outside
isakmp enable outside
isakmp policy 9 encryption 3des
ca identity abcd 192.150.50.132:cgi-bin/pkiclient.exe 192.150.50.132
ca configure abcd ra 1 100 crloptional
sysopt connection permit-IPsec
telnet timeout 5
terminal width 80
Cryptochecksum:717dd3a68ed1d1eeb297560045f88aba
: end
[OK]
```

Example #4 – #6 are based on the following topology:
These examples will assume a travelling user needs access to an inside host. The Travelling user will be using the Cisco Secure VPN Client. The travelling user will be accessing the Partner Server on the inside network 10.0.0.14. We will be using 3DES (168bit)



Example #4: In this example the System Administrator allocated 5 IP addresses to be used as virtual (internal addresses). This is done to make security tighter and access lists tight. In this example the remote user would have to enter the virtual (internal addresses) VIP manually. This example also is going to use pre-shared keys for IKE authentication (wild-card pressured key)

➤ Configuration on PIX SanJose

1. Define the hostname

```
hostname SanJose
domain-name sisu.cisco.com (optional only required in CA configurations)
```

2. Configure ISAKMP policy

```
isakmp enable outside
isakmp policy 8 authentication pre-share
isakmp policy 8 encr 3des
isakmp policy 8 hash md5
```

3. Configure static for VPN users to access inside server providing NAT

```
static (inside,outside) 171.71.2.44 10.0.0.14 netmask 255.255.255.255 0 0
```

4. Configure wild card pre-shared Key.

```
isakmp key cisco1234 address 0.0.0.0 netmask 0.0.0.0
```

5. Configure IPSec supported transforms

```
crypto IPSec transform-set strong-des esp-3des esp-sha-hmac
```

6. Create a partial access list defining remote user VIP address

```
access-list 80 permit ip host 171.71.2.44 host 192.168.15.1
access-list 80 permit ip host 171.71.2.44 host 192.168.15.2
access-list 80 permit ip host 171.71.2.44 host 192.168.15.3
access-list 80 permit ip host 171.71.2.44 host 192.168.15.4
access-list 80 permit ip host 171.71.2.44 host 192.168.15.5
```

(These VIP will also be manually entered in client see config Cisco Secure Client next to create a match)

7. Create a dynamic crypto map

```
crypto dynamic-map cisco 4 set transform-set strong-des
crypto dynamic-map cisco 4 match address 80
```

8. Define a crypto map

```
crypto map partner-map 20 IPSec-isakmp dynamic cisco
```

9. Apply the crypto map to the interface

```
crypto map partner-map interface outside
```

10. Tell PIX to Implicitly trust IPSec traffic

```
sysopt connection permit-IPSec
```

PIX Configuration

```

PIX Version 5.0(0)203
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging timestamp
no logging standby
logging console debugging
no logging monitor
no logging buffered
no logging trap
logging facility 20
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
ip address dmz 192.168.101.1 255.255.255.0
no failover
arp timeout 14400
global (outside) 1 171.71.2.45-171.71.2.50
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 171.71.2.44 10.0.0.14 netmask 255.255.255.255 0 0
access-list 80 permit ip host 171.71.2.44 host 192.168.15.1
access-list 80 permit ip host 171.71.2.44 host 192.168.15.2
access-list 80 permit ip host 171.71.2.44 host 192.168.15.3
access-list 80 permit ip host 171.71.2.44 host 192.168.15.4
access-list 80 permit ip host 171.71.2.44 host 192.168.15.5
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto dynamic-map cisco 4 match address 80
crypto dynamic-map cisco 4 set transform-set strong
crypto map partner-map 20 IPsec-isakmp dynamic cisco
crypto map partner-map interface outside
isakmp enable outside
isakmp key cisco1234 address 0.0.0.0 netmask 0.0.0.0
isakmp policy 8 authentication pre-share
isakmp policy 8 encryption 3des
isakmp policy 8 hash md5
sysopt connection permit-IPsec

```

➤ Configuration on the Cisco Secure Client

1. Select Start | Programs | Cisco Secure VPN Client | Security Policy Editor
2. Select Options | Secure | Specified Connections
3. Select Other Connection and choose Non-Secure
4. **Select Options | Global Policy Settings**
5. **Check the box marked “Allow to Specify Internal Network Address”**
6. Select File | New Connection. Rename New Connection to say toCisco. Select toCisco and select the entries as given below.
7. Connection Security: Secure
8. ID Type: IP Address 171.71.2.44
9. Port: grayed out
10. Protocol All
11. Connect using Secure Gateway Tunnel: Check the box
12. ID_Type: IP Address 171.71.2.40
13. Expand toCisco and select My Identity
14. Select certificate: none
15. ID Type: IP address.
16. **Internal Network IP Address: 192.168.15.3**
(Matches on of the IP Address in access list we entered in PIX)
17. Port: All
18. Pre-shared : cisco1234
19. Select Security Policy
20. Phase 1 Negotiation: Main Mode
21. Replay Detection: Checked
22. Select Security Policy | Phase 1 | Create new proposal
23. Enter the following values for proposal 1 [IKE proposals]
24. Authentication Method: pre-share
25. Encrypt Alg: 3DES
26. Hash Alg: MD5
27. SA Life: Unspecified
28. Key Group: Diffie-Hellman Group 1
29. Select Key Exchange and create a Phase 2 Proposal
30. Encapsulation Protocol is checked
31. Encryption Alg: 3DES

- 32. Hash Alg: SHA
- 33. Encapsulation: Tunnel
- 34. Save the Policies

Snapshot of viewlog

```
15:05:18.498 toCisco - Deleting IKE SA
15:06:10.433 toCisco - SENDING>>>> ISAKMP OAK MM (SA)
15:06:11.104 toCisco - RECEIVED<<< ISAKMP OAK MM (SA)
15:06:11.384 toCisco - SENDING>>>> ISAKMP OAK MM (KE, NON)
15:06:11.785 toCisco - RECEIVED<<< ISAKMP OAK MM (KE, NON, VID)
15:06:11.975 toCisco - SENDING>>>> ISAKMP OAK MM *(ID, HASH)
15:06:12.176 toCisco - RECEIVED<<< ISAKMP OAK MM *(ID, HASH)
15:06:12.226 toCisco - SENDING>>>> ISAKMP OAK QM *(HASH, SA, NON, ID, ID)
15:06:13.538 toCisco - RECEIVED<<< ISAKMP OAK QM *(HASH, SA, NON, ID, ID,
NOTIFY:STATUS_RESP_LIFETIME)
15:06:13.588 toCisco - SENDING>>>> ISAKMP OAK QM *(HASH)
15:06:13.708 toCisco - Loading IPsec SA keys...
15:06:13.708
```

Example #5: In this example we add AAA Authentication and Authorization because we are only using a pre-shared key for IKE Authentication. This option now improves our Authentication process by adding a user name password prompt for each VPN user. We also have added the ip local pool command (IOS equivalent command is mode config). This Command eliminates the need to hard code the virtual IP addresses. All remote VPN users will have an address dynamically assigned to them out of the pool.

*****This example has several advantages' the reader should note*****

The configuration for every client is the same. Thus the same client configuration file could be sent to all remote users making the remote install much easier. You can also lock the configuration file so the remote user does cannot change or corrupt it.

Pre-share keys are being used for IKE. This allows the user to authorize an IKE without requesting the certificate for every user. If Certificates are desired for authentication they still can be used.

Strong Authentication is still being used: Every remote user is still being required to enter a user name and password by the AAA Server (Cisco Secure ACS for NT) on the DMZ. Thus even a one-time password product like Secure ID could be used

Password protection/encryption: Note the password is not being sent clearly. The password is being sent after the 3DES tunnel is started and has been 168 Bit DES encrypted.

1. Define the hostname

```
hostname SanJose
domain-name sisu.cisco.com (optional only required in CA configurations)
```

2. Configure static for VPN users to access inside server providing NAT

```
static (inside,outside) 171.71.2.44 10.0.0.14 netmask 255.255.255.255 0 0
```

3. Configure ISAKMP policy

```
isakmp enable outside
isakmp policy 8 authentication pre-share
isakmp policy 8 encr 3des
isakmp policy 8 hash md5
```

4. Configure wild card pre-shared Key.

```
isakmp key cisco1234 address 0.0.0.0 netmask 0.0.0.0
```

5. Configure IPSec supported transforms

```
crypto IPSec transform-set strong-des esp-3des esp-sha-hmac
```

6. Create a partial access list

```
access-list 80 permit ip host 171.71.2.44 host 192.168.15.1
access-list 80 permit ip host 171.71.2.44 host 192.168.15.2
access-list 80 permit ip host 171.71.2.44 host 192.168.15.3
access-list 80 permit ip host 171.71.2.44 host 192.168.15.4
access-list 80 permit ip host 171.71.2.44 host 192.168.15.5
      (match static above)      (match local pool below)
```

7. Create a dynamic crypto map

```
crypto dynamic-map cisco 4 set transform-set strong
crypto dynamic-map cisco 4 match address 80
```

8. Define a crypto map

```
crypto map partner-map 20 IPSec-isakmp dynamic cisco
```

9. Configure mode config related parameters

```
ip local pool dealer 192.168.15.1-192.168.15.5
crypto map partner-map client configuration address initiate
isakmp client configuration address-pool local dealer outside
```

(All 3 commands required for Cisco Secure VPN Client using mode config)

10. Define AAA related parameters

```
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server partnerauth protocol tacacs+
aaa-server partnerauth (dmz) host 192.168.101.2 abcdef timeout 5
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.1 255.255.255.255
partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.2 255.255.255.255
partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.3 255.255.255.255
partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.4 255.255.255.255
partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.5 255.255.255.255
partnerauth
```

(Note we are matching the mode config addresses and thus making the AAA Server only prompt the VPN users for AAA Authentication)

More AAA functions you could add:

By adding accounting to this example, the CiscoSecure ACS could provide very helpful usage reports on VPN users, and command AAA accounting for any inbound traffic.

By adding authorization we could also add more security by further restricting remote VPN users access. Enable or disable TACACS+ user authorization for services. The authentication server now also determines what services the user is authorized to access.

11. Apply the crypto map to the interface

```
crypto map partner-map interface outside
```

12. Tell PIX to Implicitly trust IPSec traffic

```
sysopt connection permit-IPSec
```

PIX Configuration

```

Building configuration...
: Saved
:
PIX Version 5.0(0)203
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging timestamp
no logging standby
logging console debugging
no logging monitor
no logging buffered
no logging trap
logging facility 20
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
ip address dmz 192.168.101.1 255.255.255.0
ip local pool dealer 192.168.15.1-192.168.15.5
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
failover ip address dmz 0.0.0.0
arp timeout 14400
global (outside) 1 171.71.2.41-171.71.2.45
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 171.71.2.44 10.0.0.14 netmask 255.255.255.255 0 0
access-list 80 permit ip host 171.71.2.44 host 192.168.15.1
access-list 80 permit ip host 171.71.2.44 host 192.168.15.2
access-list 80 permit ip host 171.71.2.44 host 192.168.15.3
access-list 80 permit ip host 171.71.2.44 host 192.168.15.4
access-list 80 permit ip host 171.71.2.44 host 192.168.15.5
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
no rip dmz passive
no rip dmz default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius

```

```
aaa-server partnerauth protocol tacacs+
aaa-server partnerauth (dmz) host 192.168.101.2 abcdef timeout 5
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.1 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.2 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.3 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.4 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.5 255.255.25
5.255 partnerauth
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto dynamic-map cisco 4 match address 80
crypto dynamic-map cisco 4 set transform-set strong
crypto map partner-map 20 IPsec-isakmp dynamic cisco
crypto map partner-map client configuration address initiate
crypto map partner-map client configuration address respond
crypto map partner-map interface outside
isakmp enable outside
isakmp key cisco1234 address 0.0.0.0 netmask 0.0.0.0
isakmp client configuration address-pool local dealer outside
isakmp policy 8 authentication pre-share
isakmp policy 8 encryption 3des
isakmp policy 8 hash md5
sysopt connection permit-IPsec
telnet timeout 5
terminal width 80
Cryptochecksum:e71669eeab8f1fb294e13ba48bbcc5f0
: end
```

Configuration on the Cisco Secure VPN Client

1. Select Start | Programs | Cisco Secure VPN Client | Security Policy Editor
2. Select Options | Secure | Specified Connections
3. Select Other Connection and choose Non-Secure
4. Select File | New Connection. Rename New Connection to say toCisco. Select toCisco and select the entries as given below.
5. Connection Security: Secure
6. ID Type: IP Address 171.71.2.44
7. Port: grayed out
8. Protocol All
9. Connect using Secure Gateway Tunnel: Check the box
10. ID_Type: IP Address 171.71.2.40
11. Expand toCisco and select My Identity
12. Select certificate: none
13. ID Type: IP address.
14. Port: All
15. Pre-shared : cisco1234
16. Select Security Policy
17. Phase 1 Negotiation: Main Mode
18. Replay Detection: Checked
19. Select Security Policy | Phase 1 | Create new proposal
20. Enter the following values for proposal 1 [IKE proposals]
21. Authentication Method: pre-share
22. Encrypt Alg: 3DES
23. Hash Alg: MD5
24. SA Life: Unspecified
25. Key Group: Diffie-Hellman Group 1
26. Select Key Exchange and create a Phase 2 Proposal
27. Encapsulation Protocol is checked
28. Encryption Alg: 3DES
29. Hash Alg: SHA
30. Encapsulation: Tunnel
31. Save the Policies

Snapshot of viewlog

```

15:33:40.075 toCisco - Deleting IKE SA
15:33:44.862 toCisco - SENDING>>>> ISAKMP OAK MM (SA)
15:33:45.533 toCisco - RECEIVED<<<< ISAKMP OAK MM (SA)
15:33:45.823 toCisco - SENDING>>>> ISAKMP OAK MM (KE, NON)
15:33:46.224 toCisco - RECEIVED<<<< ISAKMP OAK MM (KE, NON, VID)
15:33:46.414 toCisco - SENDING>>>> ISAKMP OAK MM *(ID, HASH)
15:33:46.615 toCisco - RECEIVED<<<< ISAKMP OAK MM *(ID, HASH)
15:33:46.665 toCisco - SENDING>>>> ISAKMP OAK QM *(HASH, SA, NON, ID, ID)
15:33:46.725 toCisco - RECEIVED<<<< ISAKMP OAK TRANS *(HASH, ATTR)
15:33:46.725 toCisco - Received Private IP Address = IP ADDR=192.168.15.1
15:33:46.775 toCisco - SENDING>>>> ISAKMP OAK TRANS *(HASH, ATTR)
15:33:48.087 toCisco - RECEIVED<<<< ISAKMP OAK QM *(HASH, SA, NON, ID, ID,
NOTIFY:STATUS_RESP_LIFETIME)
15:33:48.137 toCisco - SENDING>>>> ISAKMP OAK QM *(HASH)
15:33:48.257 toCisco - Loading IPsec SA keys...
15:33:48.257

```

Example # 5 Continued: Now add NT Server access:

- The client wants to log into an NT Server on the inside 10.0.0.4. In this example we will solve WINS issues by using an external WINS Server. **Important note: This is a limitation in the 5.0.1 code because we are terminating IPsec on the outside interface of the PIX. In the 5.0.2 code this will be corrected and IPsec will be allowed to terminate on the inside**
- Basic Steps:
- On a Remote PC: You can setup your WINS information on the remote PC manually by going to the TCP/IP configuration.
- Dial the ISP, ISP issues dynamic address. Also make sure Client for Microsoft Networks is enabled. Client next authenticates with the ISP (via PAP or CHAP for example)
- Using an external WINS server, the VPN client can resolve statically entered NETBIOS names that point to the outside static global IP addresses of the PIX Firewall. You should have the Windows NT domain name and the PDC (Primary Domain Controller) added statically into WINS for Windows NT authentication to resources. The internal WINS server cannot be used since it sends back the 10.0.x.x address. This address is not accessible because there are not global static command statements for a 10.0.x.x address.
- **Please see Chapter 4 in the PIX Guide for configuration and other examples of WINS with all positives and negatives listed. Please note all NT work around are not necessary if running the 5.0.2 release. You can terminate on the inside and make user part of NT Network. NO static would be required.**

- **Example #6: Client to PIX using Verisign Digital Certificates**

➤ Configuration on PIX San Jose

1. Define the hostname

hostname SanJose

2. Define the Domain Name (required in CA configurations)

domain-name sisu.cisco.com

3. Create a Net Static

static (inside,outside) 171.71.2.44 10.0.0.14

4. Configure ISAKMP policy

isakmp enable outside
isakmp policy 8 auth rsa-signature

5. Define Verisign related enrollment commands.

The IP address of onsiteIPSec.verisign.com server is 205.139.94.230

ca identity sisu.cisco.com 205.139.94.230

ca configure sisu.cisco.com ca 1 20 crloptional

[where 1 is retry period and 20 is retry count. crl optional says no crl checking]

6. Generate RSA key pair

ca generate rsa key 1024

7. Get the public key and the certificate of the CA server

ca authenticate sisu.cisco.com

8. Contact your CA Administrator and send the your certificate request.

ca enroll sisu.cisco.com cisco

[where cisco is a challenge password. Can be anything]

9. Configure IPSEC supported transforms

crypto IPsec transform-set basic-des esp-des esp-md5-hmac

10. Save Keys, certificates and CRL's in flash. Very Important

ca save all

11. Create a partial access list

access-list 80 permit ip 10.0.0.0 255.0.0.0 192.168.12.0 255.255.255.0

12. Define a crypto map

crypto map toNewYork 20 IPsec-isakmp

crypto map toNewYork 20 match address 80

crypto map toNewYork 20 set transform-set basic

crypto map toNewYork 20 set peer 192.150.50.51

13. Apply the crypto map to the interface

crypto map toNewYork interface outside

14. Tell PIX to Implicitly trust IPSec traffic

sysopt connection permit-IPSec

```

Building configuration...
: Saved
:
PIX Version 5.0(0)203
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SanJose
domain-name sisu.cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging timestamp
no logging standby
logging console debugging
no logging monitor
no logging buffered
no logging trap
logging facility 20
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 171.71.2.40 255.255.255.224
ip address inside 10.0.0.1 255.0.0.0
ip address dmz 192.168.101.1 255.255.255.0
ip local pool dealer 192.168.15.1-192.168.15.5
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
failover ip address dmz 0.0.0.0
arp timeout 14400
global (outside) 1 171.71.2.41-171.71.2.45
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 171.71.2.44 10.0.0.14 netmask 255.255.255.255 0 0
access-list 80 permit ip host 171.71.2.44 host 192.168.15.1
access-list 80 permit ip host 171.71.2.44 host 192.168.15.2
access-list 80 permit ip host 171.71.2.44 host 192.168.15.3
access-list 80 permit ip host 171.71.2.44 host 192.168.15.4
access-list 80 permit ip host 171.71.2.44 host 192.168.15.5
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
no rip dmz passive
no rip dmz default
route outside 0.0.0.0 0.0.0.0 171.71.2.33 1
timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00 udp 0:02:00

```

```

timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server partnerauth protocol tacacs+
aaa-server partnerauth (dmz) host 192.168.101.2 abcdef timeout 5
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.1 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.2 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.3 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.4 255.255.25
5.255 partnerauth
aaa authentication any inbound 10.0.0.14 255.255.255.255 192.168.15.5 255.255.25
5.255 partnerauth
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
crypto IPsec transform-set strong esp-3des esp-sha-hmac
crypto dynamic-map cisco 4 match address 80
crypto dynamic-map cisco 4 set transform-set strong
crypto map partner-map 20 IPsec-isakmp dynamic cisco
crypto map partner-map client configuration address initiate
crypto map partner-map interface outside
isakmp enable outside
isakmp key cisco1234 address 0.0.0.0 netmask 0.0.0.0
isakmp client configuration address-pool local dealer outside
isakmp policy 5 encryption 3des
isakmp policy 5 group 2
isakmp policy 8 authentication pre-share
isakmp policy 8 encryption 3des
isakmp policy 8 hash md5
ca identity sisu.cisco.com 205.139.94.230:cgi-bin/pkiclient.exe
ca configure sisu.cisco.com ca 1 50 crloptional
sysopt connection permit-IPsec
telnet timeout 5
terminal width 80
Cryptochecksum:c3572b3abf418020dd1d3cb68106f836
: end

```

Example 5 continued: Now Configure Cisco Secure VPN Client to receive Digital Certificate from the Verisign CA Server

A. Get the Root CA Certificate using CEP

1. Select Start | Program | SafeNet Soft-PK | Certificate Manager. The Certificate Manager window displays.
1. Select CA Certificates
2. Click on Retrieve CA Certificate
3. CA Domain: sisu.cisco.com
4. On-line Certificate Server: <http://onsiteIPsec.verisign.com/cgi-bin/pkiclient.exe>
5. Click Ok

6. Click View and verify to confirm that certificate is valid

B. Create Public/Private Key pair and request the certificate for the client from the Verisign CA Server.

1. Select My Certificates in the Certificates Manager
2. Select Request Certificate
3. Fill in the following

Name: Give a Unique name

IP Address: Do not fill the IP address field.

Email : Give your email ID: judy@cisco.com

Domain name: chicago.sisu.cisco.com where chicago is the machine name

Challenge Phrase: Give any password

4. Select On-line enrollment method
5. Select OK. The client will generate public/private key pairs.
6. You should see the following message: *“Your certificate has been accepted. Certificate issuance is pending approval by your Certificate Authority Administrator.”*
7. Call the CA Administrator and inform about your pending request.
8. CA Administrator approves the request, click Certificate Requests on the main tab. Select your pending certificate and click Retrieve.
9. When prompted, add the personal certificate
- 10.** Click on My Certificates and view your certificate
11. Select view and verify to confirm your digital certificate.
12. Close the CA Certificate Manager Window.

Crypto Vocabulary

Authentication Header (AH): A security protocol that provides authentication and optional replay-detection services for IP datagrams. The Authentication Header (AH) may appear after any other headers, which are examined at each hop, and before any other headers which are not examined at an intermediate hop. The IPv4 or IPv6 header immediately preceding the Authentication Header will contain the value 51 in its Next Header (or Protocol) field. Refer to the RFC 2402 for details.

Authentication Hashes - used as cryptographic checksums to determine if data has been modified in transit.

- **MD5 (Message Digest 5)**

One way hash that combines a shared secret and the message (the header and payload), to produce a 128-bit value. The recipient of the message runs the same hash of the message and compares it with the inserted hash value to yield (hopefully) the same result. This indicates that nothing in the packet has been changed in transit.

- **SHA (Secure Hash Algorithm)**

Similar to MD5 but produces a 160 bit hash value. Less chance of collisions but takes longer to calculate than MD5.

- **HMAC:** A mechanism for message authentication using cryptographic hashes such as SHA and MD5. For an exhaustive discussion of HMAC, check out RFC 2104.

Certification Authority (CA): A third-party entity that is responsible for issuing and revoking certificates. Each device that has its own certificate and public key of the CA can authenticate every other device within a given CA's domain. A certificate authority is analogous to DMV which issues and revokes driver licenses after doing some out of band checking. The driver licenses are then used to prove identity whenever needed.

Certificate: A cryptographically signed object that contains an identity and a public key associated with this identity. It is like a driver license.

Certificate Revocation List (CRL): A list of digital certificates that are revoked by a given CA. This is analogous to a list of stolen revoked driver licenses. This list is automatically downloaded. It can also be downloaded manually.

Crypto Map: A PIX or IOS configuration entity that performs two primary functions: (1) it selects data flows that need security processing and (2) it defines the policy for these flows and the crypto peer that traffic needs to go to. A crypto map is applied to an interface. Two different types of crypto map can be defined: regular crypto maps and dynamic crypto maps.

1. A **regular crypto map** is used when the IP address of the peer is definable beforehand, this map is characterized by the inclusion of the "set peer" statement.
2. A **dynamic crypto map** is used when the identity of the remote peer is not known beforehand. In this case, as long as the router can successfully authenticate the remote, the IP address of the remote will be used as the IPSec peer point, so the local router learns the peer's IP address dynamically. This is ideal for remote users. The only caveat with this map is that the local router cannot be the initiator of the SA establishment process (unless Tunnel Endpoint Discovery is configured). An access list is also optional.

Data integrity: Data integrity mechanisms, through the use of secret-key based or public-key based algorithms, that allow the recipient of a piece of protected data to verify that the data has not been modified in transit.

Data confidentiality: Method where protected data is manipulated so that no attacker can read it. This is commonly provided by data encryption and keys that are only available to the parties involved in the communication.

Data origin authentication: A security service where the receiver can verify that protected data could have originated only from the sender. This service requires a data integrity service plus a key distribution mechanism, where a secret key is shared only between the sender and receiver.

Data Encryption Standard (DES): The National Bureau of Standards published the DES algorithm in 1977 and it is a secret key encryption scheme based on the Lucifer algorithm from IBM. It has several flavors. PIX supports DES-CBC flavor.

DES-CBC (Cipher Block Chaining) .

Requires an Initialization Vector (IV) to begin the encryption process. Take a 64 bit block of message, XOR it with the IV, then encrypt the result with the key. The result is a 64 bit block of ciphertext that is fed or chained via an XOR with the next 64 bits of the message.

Diffie-Hellman: A method of establishing a shared key over an insecure medium. This shared key is used to generate the encryption keys for data encryption (IPSEC).

Diffie-Hellman Example

Host A

prime $p = 5$, primitive $g = 3$

Choose X_a such that

$0 \leq X_a < p$, $X_a = 2$

$Y_a = g^{X_a} \text{ mod } p$

$= 3^2 \text{ mod } 5$

$= 4$

Exchange Values

p, g, Y_a ----->

$K_e = Y_b^{X_a} \text{ mod } p$

$= 1^2 \text{ mod } 5$

$= 1$

Host B

prime $p = 5$, primitive $g = 3$

Choose X_b such that

$0 \leq X_b < p$, $X_b = 4$

$Y_b = g^{X_b} \text{ mod } p$

$= 3^4 \text{ mod } 5$

$= 1$

Exchange Values

<----- p, g, Y_b

$K_e = Y_a^{X_b} \text{ mod } p$

$= 4^4 \text{ mod } 5$

$= 1$

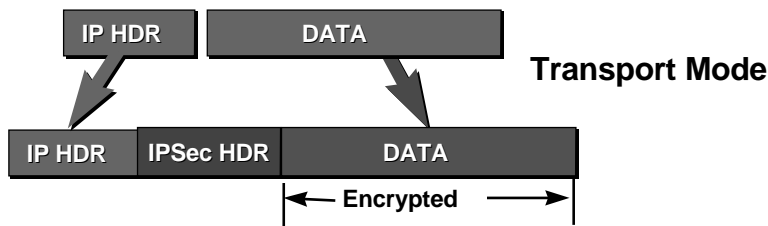
Encapsulating Security Payload (ESP):

The Encapsulating Security Payload (ESP) may appear anywhere after the IP header and before the final transport-layer protocol. The Internet Assigned Numbers Authority has assigned Protocol Number 50 to ESP. The IP Encapsulating Security Payload (ESP) seeks to provide confidentiality and integrity by encrypting data to be protected and placing the encrypted data in the data portion of the IP Encapsulating Security Payload. In Tunnel-mode ESP, the original IP datagram is placed in the encrypted portion of the Encapsulating Security Payload and that entire ESP frame is placed within a datagram having unencrypted IP headers. The information in the unencrypted IP headers is used to route the secure datagram from origin to destination. In Transport-mode

ESP, the ESP header is inserted into the IP datagram immediately prior to the transport-layer protocol header (e.g., TCP, UDP, or ICMP). In this mode bandwidth is conserved because there are no encrypted IP headers or IP options.

- **ESP in Transport mode**

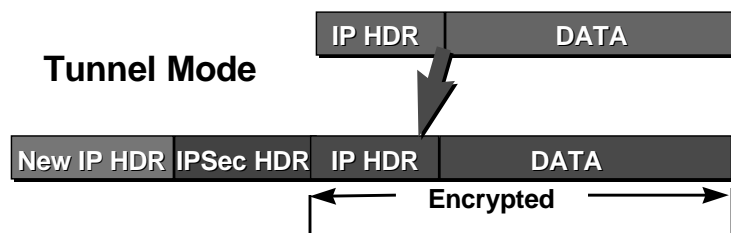
Only the IP payload is encrypted, and the original IP headers are left intact. This mode has the advantage of adding only a few bytes to each packet. It also allows devices on the public network to see the final source and destination of the packet. This capability allows you to enable special processing (for example, quality of service) in the intermediate network based on the information on the IP header. However, the Layer 4 header will be encrypted, limiting the examination of the packet.



- **ESP in Tunnel mode**

In tunnel mode, the entire original IP datagram is encrypted, and it becomes the payload in a new IP packet. This mode allows a network device, such as a router, to act as an IPSec proxy. That is, the router performs encryption on behalf of the hosts. The source's router encrypts packets and forwards them along the IPSec tunnel. The destination's router decrypts the original IP datagram and forwards it on to the destination system.

As defined by the IETF, IPSec transport mode can only be used when both the source and the destination systems understand IPSec. In most cases, you deploy IPSec with tunnel mode. Doing so allows you to implement IPSec in the network architecture without modifying the operating system or any applications on your PCs, servers, and hosts.

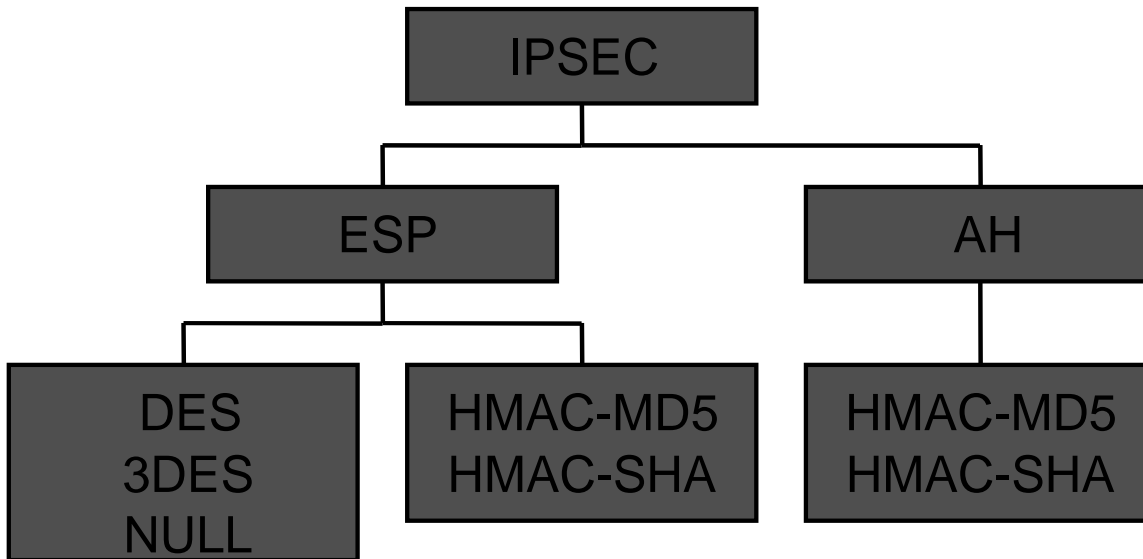


Internet Key Exchange (IKE): A hybrid protocol that uses part Oakley and part of another protocol suite called SKEME inside the Internet Security Association and Key Management Protocol (ISAKMP) framework. IKE is used to establish a shared security policy and authenticated keys for services (such as IPSec) that require keys. Before any IPSec traffic can be passed, each router/firewall/host must be able to verify the identity of its peer. This can be done by manually entering pre-shared keys into both hosts, or by a CA service. This is the protocol formerly known as ISAKMP/Oakley, and is defined in The Internet Key Exchange (IKE). A potential point of confusion is that the acronyms "ISAKMP" and "IKE" are both used in to refer to the same thing. These two items are somewhat different, as you will see in the next definition.

Internet Security Association and Key Management Protocol (ISAKMP): A protocol framework that defines the mechanics of implementing a key exchange protocol and negotiation of a security policy. ISAKMP is defined in the Internet Security Association and Key Management Protocol (ISAKMP).

- **Oakley:** A key exchange protocol that defines how to acquire authenticated keying material. The basic mechanism for Oakley is the Diffie-Hellman key exchange algorithm. You can find the standard in The OAKLEY Key Determination Protocol.

IPSEC: Provides data encryption and data integrity. You can select different combinations from the chart below:



If an AH option is used, the hash is performed on the “new IP header” and fields after AH. Often a Hash Message Authentication Code is used to strengthen the actual hash value. HMAC involves concatenation of the shared secret with the message before the hash is applied. The HMAC can also provide protection against replay attacks. By hashing the IP header (all non-changeable fields) that precedes the AH header, AH can detect any changes in the packet’s addressing information or attempts to bypass AH itself. In IPv4, the use of an AH method is not enforced.

ESP provides protection for the datagram, and can provide its own authentication service via the use of an ESP transform and a HMAC algorithm. This differs from the authentication afforded by AH in that the hash is applied to the section of the datagram starting with the ESP header.

Peer Authentication Methods

Required to authenticate the data flows between peers. Also used to generate a shared secret key to protect the IKE channel via des-cbc. This shared secret key is also used as a basis for creating the IPsec shared secret encryption key by combining it with a random value (nonce).

- **Digital Signature**

A device registers its public keys with a CA (Certificate Authority). Each device enrolls with a CA so that when required, the device may provide a certificate issued by the trusted CA to the requesting peer for authentication. The certificate binds the device’s identity with its public key. The sending peer signs the exchange (generally payload data and hash value) with its private key. The recipient decrypts the signature with the sender’s public key, this provides authentication. Using a digital signature also provides non-repudiation.

- **Pre-Shared Keys**

Shared value pre-configured for each prospective peer. Very basic.

Perfect Forward Secrecy (PFS): PFS ensures that a given IPsec SA's key was not derived from any other secret (like some other keys). In other words, if someone were to break a key, PFS ensures that the attacker would not be able to derive any other key. If PFS were not enabled, someone could hypothetically break the IKE SA secret key, copy all the IPsec protected data, and then use knowledge of the IKE SA secret to compromise the IPsec SAs setup by this IKE SA. With PFS, breaking IKE would not give an attacker immediate access to IPsec. The attacker would have to break each IPsec SA individually.

Replay-detection: A security service where the receiver can reject old or duplicate packets in order to defeat replay attacks (replay attacks rely on the attacker sending out older or duplicate packets to the receiver and the receiver thinking that the bogus traffic is legitimate). Replay-detection is done by using sequence numbers combined with authentication, and is a standard feature of IPsec.

RSA: A public key cryptographic algorithm (named after its inventors, Rivest, Shamir and Adleman) with a variable key length. RSA's main weakness is that it is significantly slow to compute compared to popular secret-key algorithms, such as DES. Cisco's IKE implementation uses a Diffie-Hellman exchange to get the secret keys. This exchange can be authenticated with RSA (or pre-shared keys). With the Diffie-Hellman exchange, the DES key never crosses the network (not even in encrypted form), which is not the case with the RSA encrypt and sign technique. RSA is not public domain, and must be licensed from RSA Data Security.

Security Association (SA): An instance of security policy and keying material applied to a data flow. Both IKE and IPsec use SAs, although SAs are independent of one another. IPsec SAs are unidirectional and they are unique in each security protocol. A set of SAs are needed for a protected data pipe, one per direction per protocol. For example, if you have a pipe that supports ESP between peers, one ESP SA is required for each direction. SAs are uniquely identified by destination (IPsec endpoint) address, security protocol (AH or ESP), and security parameter index (SPI). IKE negotiates and establishes SAs on behalf of IPsec. A user can also establish IPsec SAs manually.

An IKE SA is used by IKE only, and unlike the IPsec SA, it is bi-directional.

Transform: A transform describes a security protocol (AH or ESP) with its corresponding algorithms. For example, ESP with the DES cipher algorithm and HMAC-SHA for authentication.

Understanding Mode Configuration Feature

Assigning Internal IP Addresses to Clients

In a basic configuration the client dials into a local ISP and the ISP issues a routable address from the ISP's pool. In this case when tunneling to an IPSec gateway, the IP datagram looks something like the following:

200.200.200.6	192.168.1.1	IPSec Header	200.200.200.6	172.17.11.20	IP Payload
---------------	-------------	--------------	---------------	--------------	------------

Tunnel Header

IP Header

The source IP address of the tunnel (the client as the IPSec peer) and original IP source address are the same. Once the datagram reaches the destination IPSec peer, the tunnel and IPSec headers are removed and the cleartext original datagram remains and is forwarded. If a filtering router, or firewall must be traversed before getting to 172.17.11.20, then that firewall will need to allow the datagram based on the ISP issued address of 200.200.200.6. In other words for remote users that are issued dynamic IP addresses at random, the firewall's policy would have to allow any source address as there is no way to determine which addresses belong to trusted parties.

The way around this, is to assign an internal IP address to the client in addition to the ISP's address. This internal IP address is issued by the home gateway's administrator so there is control over which parties are assigned addresses, and firewall policy may then be based on the administered addresses :

200.200.200.6	192.168.1.1	IPSec Header	10.1.1.6	172.17.11.20	IP Payload
---------------	-------------	--------------	----------	--------------	------------

Tunnel Header

IP Header

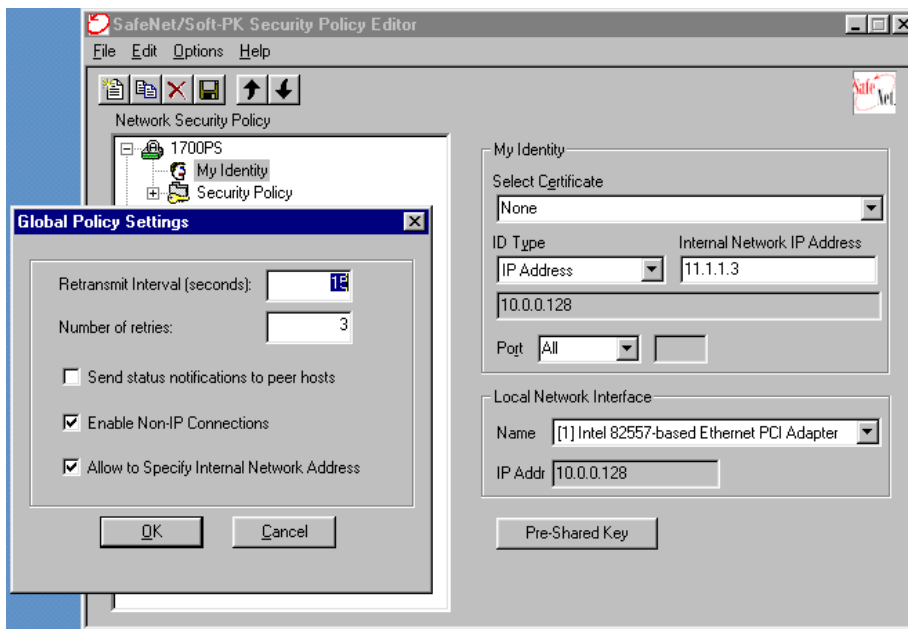
Now the firewall will allow the datagram based on the source address of 10.1.1.6, something recognizable to the corporate environment.

a) Manually assigning an Internal IP Address to the client.

Client Configuration

Use Options -> Global Policy Settings to allow the specification of an internal IP address.

Under MY Identity enter the IP address for the client. It must be an address from a unique IP address range, and this range may be private.



PIX Configuration

No specific configuration is required on the PIX except modifying the access list to accommodate internal addresses.

The major issue with this method is that the addresses must be manually added to each client.

b) Using Mode Config to Dynamically issue an IP Address to a Client.

A better way to allocate IP addresses is via Mode Configuration. Mode config is defined as an extension to IKE and occurs after IKE Main Mode and before IKE Quick Mode.

Here is the procedure applied to the PIX and the client.

1. Dial ISP using PPP via modem
2. Establish the IKE SA with gateway
3. Gateway sends ISAKMP_CFG_SET to client
4. Client sends ISAKMP_CFG_ACK
5. Client has internal attributes.

Client Configuration

Nothing specific is required. The client transparently accepts the address from the router. If static Internal IP address exists, it is superceded by the mode config issued address.

PIX Configuration

```
ip local pool dealer 192.168.15.1-192.168.15.5
crypto map partner-map client configuration address initiate
crypto map partner-map client configuration address respond
isakmp client configuration address-pool local dealer outside
```

You can view the log to see the address being accepted:

```
17:53:32.820 Mode - SENDING>>> ISAKMP OAK MM (SA)
17:53:33.100 Mode - RECEIVED<<< ISAKMP OAK MM (SA)
17:53:33.540 Mode - SENDING>>> ISAKMP OAK MM (KE, NON)
17:53:33.700 Mode - RECEIVED<<< ISAKMP OAK MM (KE, NON, CERT_REQ, VID)
17:53:33.920 Mode - SENDING>>> ISAKMP OAK MM *(ID, CERT, CERT_REQ, SIG)
17:53:34.310 Mode - RECEIVED<<< ISAKMP OAK MM *(ID, CERT, SIG)
17:53:34.580 Mode - SENDING>>> ISAKMP OAK QM *(HASH, SA, NON, ID, ID)
17:53:34.690 Mode - RECEIVED<<< ISAKMP OAK TRANS *(HASH, ATTR)
17:53:34.690 Mode - Received Private IP Address = IP ADDR=192.168.50.2
17:53:34.750 Mode - SENDING>>> ISAKMP OAK TRANS *(HASH, ATTR)
17:53:35.020 Mode - RECEIVED<<< ISAKMP OAK QM *(HASH, SA, NON, ID, ID, NOTIFY:STATUS_RESP_LIFETIME)
17:53:35.240 Mode - SENDING>>> ISAKMP OAK QM *(HASH)
17:53:35.350 Mode - Loading IPSec SA keys...
```

PIX Cryptography Command Reference:

CA Commands: The following commands allow you to configure PIX to inter-operate with a Certificate Authority Server

ca authenticate <ca_nickname> [<fingerprint>]
ca configure <ca_nickname> **ca|ra** <retry_period> <retry_count> [**crloptional**]
no ca configure <ca_nickname>
ca crl request <ca_nickname>
ca enroll <ca_nickname> <challenge_password> [serial] [ipaddress]
no ca enroll
ca generate rsa key|specialkey <key_modulus_size> <fully_qualified_domain_name>
show ca mypubkey rsa
ca zeroize rsa
ca identity <ca_nickname> <ca_ipaddress>[:<ca_script_location>][<ldap_ipaddress>]
no ca identity <ca_nickname>
show ca identity
ca save all
no ca save all
show ca certificate

IKE: The following commands are related to Internet Key Exchange (IKE)

isakmp enable <interfacename>
no isakmp enable <interfacename>
isakmp policy priority **authentication** pre-share|rsa-sig
no isakmp policy priority **authentication** pre-share|rsa-sig
isakmp policy priority **encryption** des|3des
no isakmp policy priority **encryption** des|3des
isakmp policy priority **hash** md5|sha
no isakmp policy priority **hash** md5|sha
isakmp policy priority **group** group1|group2
no isakmp policy priority **group** group1|group2
isakmp policy priority **lifetime** seconds
no isakmp policy priority **lifetime** seconds
isakmp key-string **address** peer-address **netmask** peer-netmask
no isakmp key-string **address** peer-address **netmask** peer-netmask
show isakmp policy
show isakmp sa
isakmp debug debug level
clear crypto isakmp <sa|connection_id>
isakmp identity <ipaddress|hostname>
no isakmp identity <ipaddress|hostname>

IPSec Commands: The following commands are related to IPSec

```
crypto ipsec transform-set <name> transform1 [transform2] [transform3]
crypto map <map-name> <seq-num> <IPSec-isakmp|IPSec-manual>
crypto map <map-name> <seq-num> match address <access-list number>
no crypto map <map-name> <seq-num> match address <access-list number>
crypto map <map-name> <seq-num> set peer ip-address1 [ip-address2] [..ip-address40]
no crypto map <map-name> <seq-num> set peer ip-address1 [ip-address2] [..ip-address40]
crypto map <map-name> <seq-num> set pfs <group1 | group 2>
no crypto map <map-name> <seq-num> set pfs <group1 | group 2>
crypto map <map-name> <seq-num> set security-association lifetime < seconds seconds | kilobytes kilobytes>
no crypto map <map-name> <seq-num> set security-association lifetime < seconds seconds | kilobytes kilobytes>
crypto map <map-name> <seq-num> set transform-set transform-set-name1 [..name9]
crypto map <map-name> <seq-num> match address <access-list-number>
no crypto map <map-name> <seq-num>
```

Debug: The following commands are related to debugging

```
debug crypto isakmp
debug crypto IPSec
debug crypto ca
```

Show: The following show commands are related to cryptography.

```
show crypto isakmp policy
show crypto isakmp sa
show crypto IPSec sa
```

Clear: The following clear commands are related to cryptography.

```
clear crypto IPSec
clear crypto isa sa
```



Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems Europe s.a.r.l.
Parc Evolic, Batiment L1/L2
16 Avenue du Quebec
Villebon, BP 706
91961 Courtabouff Cedex
France
<http://www-europe.cisco.com>
Tel: 33 1 6918 6100
Fax: 33 1 6928 8326

**Americas
Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-7660
Fax: 408 527-0883

Asia Headquarters
Nihon Cisco Systems K.K.
Fuji Building, 9th Floor
3-2-3 Marunouchi
Chiyoda-ku, Tokyo 100
Japan
<http://www.cisco.com>
Tel: 81 3 5219 6250
Fax: 81 3 5219 6001

**Cisco Systems has more than 200 offices in the following countries. Addresses, phone numbers, and fax numbers are listed on the
Cisco Connection Online Web site at <http://www.cisco.com>.**

Argentina • Australia • Austria • Belgium • Brazil • Canada • Chile • China (PRC) • Colombia • Costa Rica • Czech Republic • Denmark
England • France • Germany • Greece • Hungary • India • Indonesia • Ireland • Israel • Italy • Japan • Korea • Luxembourg • Malaysia
Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal • Russia • Saudi Arabia • Scotland • Singapore
South Africa • Spain • Sweden • Switzerland • Taiwan, ROC • Thailand • Turkey • United Arab Emirates • United States • Venezuela

Copyright © 1998 Cisco Systems, Inc. All rights reserved. Printed in USA. AccessPath, AtmDirector, Cache Director System, the CCIE logo, CD-PAC, Centri, Centri Bronze, Centri Gold, Centri Security Manager, Centri Silver, the Cisco Capital logo, Cisco IOS, the Cisco IOS logo, Cisco Link, the Cisco Powered Network logo, the Cisco Press logo, ClickStart, ControlStream, Fast Step, FragmentFree, IGX, JumpStart, Kernel Proxy, LAN-LAN Enterprise, LAN-LAN Remote Office, MICA, Natural Network Viewer, NetBeyond, Netsys Technologies, Packet, PIX, Point and Click Internetworking, Policy Builder, RouteStream, Secure Script, SMARNet, StrataSphere, StrataSphere BILLder, StrataSphere Connection Manager, StrataSphere Modem, StrataSphere Optimizer, Stratum, StreamView, SwitchProbe, The Cell, TokenSwitch, TrafficDirector, VirtualStream, VlanDirector, Workgroup Director, Workgroup Stack, and XCI are trademarks; The Network Works No Excuses is a service mark; and BPX, Catalyst, Cisco, Cisco Systems, the Cisco Systems logo, EtherChannel, FastHub, FastPacket, ForeSight, IPX, LightStream, OptiClass, Phase/IP, StrataCom, and StrataView Plus are registered trademarks of Cisco Systems, Inc. in the U.S. and certain other countries. All other trademarks mentioned in this document are the property of their respective owners. 97 11R