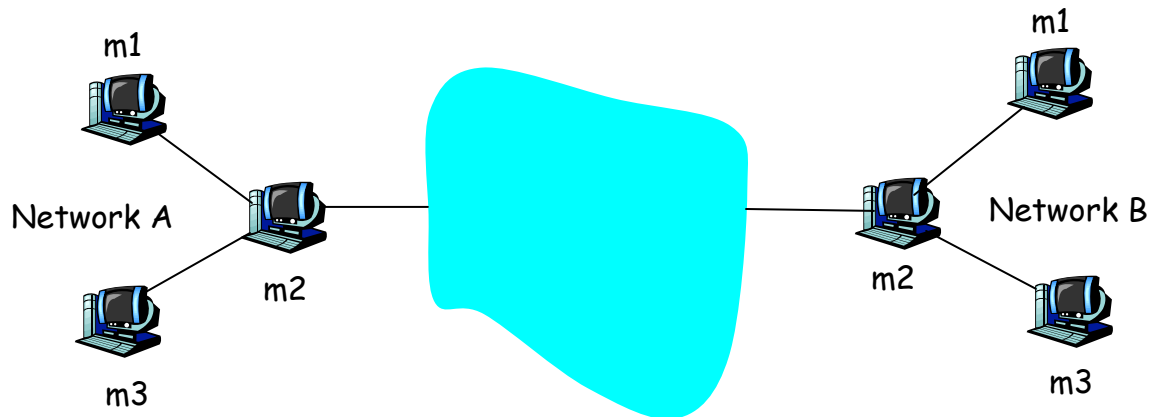


IPSec Lab



Ideally in this lab you will have access to six linux boxes, with two of boxes having three interface cards and serving as a router. If you have only two linux boxes, you can do a scaled back version of lab using only the transport mode between the two linux boxes. The lab described below assumes you create the full set-up, as shown in the above figure.

Part 1:IPSec connection with Manual Keying in the same network (Transport mode)

Throughout this lab, you will be using the ESP protocol (rather than the AH protocol). In this part, you will set up IPsec SAs between two hosts m1 and m3 (transport mode) in one of the networks. All traffic will pass through m2. Using m2, you will test your SAs by sniffing and examining the packets sent between the two hosts. You may use any sniffer you want (e.g., Wireshark, Snort, or tcpdump).

To create your SAs, you will need to edit the setkey configuration files in m1 and m3. To edit the configuration file with vi, type `vi /etc/setkey.conf`.

Answer the following questions:

1. Explain, in detail, how you configured and setup the IPsec SAs. Include your configuration file in your report (*setkey.conf* file).
2. What features of IPsec are being used in this example? Confidentiality? Integrity? Authentication? What options in the ESP protocol are being used? What crypto algorithms are being used?

Examine one of the IPsec encrypted packets. Answer the following questions:

3. Is the IP header encrypted? How large is the IP header. Is the entire IP payload encrypted?
4. What is the protocol number for ESP?
5. Can you tell whether the datagram is carrying UDP, TCP, or ICMP data? How?
6. What is the SPI for this SA from your host to your partner's host? What is the SPI for this SA from your partner's host to your host?
7. How are the sequence numbers changing in each of the SAs?
8. In your own words describe in what cases is manual keying feasible and in what cases it is not.

Support your solutions with screenshots of sniffed traffic where required.

Before you begin the lab you might want to review the Power Point lecture and gather more information about IPSec and its implementation methods. Some useful websites are:

- <http://www.ipsec-howto.org/x304.html>. This document provides several examples, many of which you'll find very useful in this lab.
- <http://www.unixwiz.net/techtips/iguide-ipsec.html>. This is a nice site which takes a graphical approach to explaining IPSec the intricacies in it.
- <http://www.slackbasics.org/html/ipsec.html> Another useful site.

Part 2: IPSec connection with Manual Keying between hosts in different networks (Tunnel Mode)

In this part of the lab you will setup IPSec security associations manually using the setkey.conf file like in the previous part, but this time the communicating hosts will be communicating via their respective routers, and you will have to set up IPSec in the tunnel mode between the m2s of both networks.

You are required to keep the following instructions in mind.

- To capture the traffic, please make sure you run the sniffer on the correct interface. By default, tcpdump captures packets from the first interface and that's veth0 in M2.
- If your networks are NATed, you will have to remove the NATs (see Appendix).

After setting up the SAs, answer the following questions:

1. Explain in detail how you set up the SAs and include your configuration file in your report.
2. What is IPsec doing for you in this example? Confidentiality? Integrity? Authentication? What options in the ESP protocol are being used? What crypto algorithms are being used?

Examine the captured packets and answer the following:

3. Is the IP header encrypted? How large is the IP header. Is the entire IP payload encrypted?
4. If the capture was properly done on M2 you should be able to see ESP packets in both directions but the ICMP packet in only one direction. Why?
5. If ESP authentication was used, how would it be different from the extent of authentication provided by using AH in transport mode ?

Part 3: IPSec connection with Manual Keying between hosts in different subnets (Transport Mode):

In this part of the lab you will setup IPSec security associations manually using the setkey.conf file like in Part2, but you are to use the transport mode this time and the SA should be set up between M1 of one subnet to the M1/M3 of the other subnet.

After setting up the SA, answer the following questions.

1. Explain in detail how you set up the connection and include your configuration file in your report.
2. What is IPsec doing for you in this example? Confidentiality? Integrity? Authentication? What options in the ESP protocol are being used? What crypto algorithms are being used?

Examine the captured packets and answer the following:

3. Is the IP header encrypted? How large is the IP header. Is the entire IP payload encrypted?
4. Can you tell whether the datagram is carrying UDP, TCP, or ICMP data?How?
5. Compare the transport mode to the tunnel mode and list the differences that you find.
6. What are the source and destination IP addresses in the captured packets? Are they the same as they were in the tunnel mode?

Support your answers with relevant screen shots.

7. In your setkey.conf file you can use one of the following levels for establishing an IPSec connection: “use”, “require”. How are they different? Will you be able to enable IPSec protection if “use” was used instead of “require”? . What are the other options?

Part 4 : Automatic Keying using IKE with preshared secret (Transport Mode)

Repeat Part 3, but this time do not manually install the shared keys. Instead, we will manually install a preshared secret (unfortunately called a preshared key in IKE jargon) in the two hosts. We will then run IKE, which will use these preshared secrets to generate the encryption and authentication keys automatically.

The connection should be established between an internal host of one group and an internal

host of the other, and it should be done using the transport mode. For this task you will need to use Racoon, the IKE daemon. In addition to the setkey.conf file you will have to create a racoon.conf on each of the two hosts. For each host, you will also need to create a preshared key file, which has exactly one line with the IP address of the remote host and the preshared secret (a string of any length).

Instructions:

- The racoon.conf file is read by racoon. You can set the sainfo block to anonymous.
- The file psk.txt will hold the remote host's identity and the shared secret. You have to make sure that this file can be read only by privileged users and should use the chmod command to change permissions to 400 or 600 . Racoon will not run properly if this is not the case.
- Use the IKE main mode.
- The setkey.conf file should not contain the SA block as generating the SA (including the encryption keys) is the job of IKE.
- Make sure to flush the previous SAD and SPD before the new ones are loaded.
- An IKE session will only be setup if a connection is initiated. Use the PING command to do this.
- You can run racoon in the foreground for diagnostics, using the command

racoon -F -f/path/to/racoon.conf

After setting up the connection, answer the following questions.

1. Explain in detail how you set up the connection and include your configuration files in your report.
2. What is IPsec doing for you in this example? Confidentiality? Integrity? Authentication? What options in the ESP protocol are being used? What crypto algorithms are being used?
3. The IPSec SA establishment takes place in two phases. The identity protection mode and the quick mode. What are they? Explain each message and provide screenshots for each of them.
4. How are the ISAKMP SA and IKE SA related?
5. What are the SPIs in each direction.

Part 5: Automatic Keying with IKE using certificates (Tunnel Mode)

Setup the IPsec connection like in part 4 using IKE, but this time with certificates and in the tunnel mode. Like in part 2, the tunnel should be setup between the two routers (M2s) .

Instructions:

- Use OpenSSL to generate certificates and the private key.
- This time set the sainfo block with M2s addresses.

After setting up the connection, answer the following questions.

1. Explain in detail how you set up the connection and include your configuration files in your report.
2. What is IPsec doing for you in this example? Confidentiality? Integrity? Authentication? What options in the ESP protocol are being used? What crypto algorithms are being used?
3. What is perfect forward secrecy and what does it do? Are you using it?
4. How is the key generation different in certificates than with preshared keys? Which one is more feasible in the real world?
5. What are the SPIs in each direction?
6. Will the same configuration protect traffic flowing between the other two internal hosts ?

Appendix

It is not possible to set up SAs between a node in one network to a node in another network when NAT is being used. So NAT must be removed and static routes must be configured between the two networks to enable communication between internal nodes.

Here is a possible solution to overcome the problem of NAT:

1. You first decide on the subnets for their internal network. For example: Network 1's M1-M2 subnet could be [192.168.10.0](#) and M3-M2 subnet could be [192.168.11.0](#) and Network 2's M1-M2 subnet could be [192.168.12.0](#) and M3-M2 subnet could be [192.168.13.0](#)
2. Remove NATing by typing *iptables -t nat -F*
3. Both networks set M2's veth1 and veth2 IP, M1's eth0 and M3's eth0 ip to match the subnet change. For example team 1 M2's veth1 should be changed from [192.168.100.1](#) to [192.168.10.1](#) ...using command:
ifconfig veth1 192.168.10.1 netmask 255.255.255.0
4. Add static routes in M2.
For network 1's M2 one should type

route add -net [192.168.12.0/24](#) gw <team 2's M2's external IP>
route add -net [192.168.13.0/24](#) gw <team 2's M2's external IP>

and in network 2's M2

route add -net [192.168.10.0/24](#) gw <team 1's M2's external IP>
route add -net [192.168.11.0/24](#) gw <team 1's M2's external IP>
5. You should set the default router in M1, M2 and M3 correctly after changing its IP. To do that type the *route add default gw <gw's ip>* For example: For M1 of network 1 you should type

route add default gw [192.168.10.1](#) and for m3:

route add default gw [192.168.11.1](#).

Please note that for M2 the default router should be [10.24.100.1](#).