

Group C- Assignment No. 17

Title:

To study the IPsec (ESP and AH) protocol by capturing the packets using Wireshark tool.

Outcomes:

Retrieve IPsec (ESP and AH) protocol by capturing the packets using Wireshark tool.

Theory:

The IP security (IPSec) is an Internet Engineering Task Force (IETF) standard suite of protocols

~~between two communicating hosts. The~~ protocols needed for secure key exchange and key management are defined in it.

What Ports Does IPSEC Operate On?

UDP port 500 should be opened as should IP protocols 50 and 51. UDP port 500 should be opened

to allow for ISAKMP to be forwarded through the firewall while protocols 50 and 51 allow ESP and

AH traffic to be forwarded resp

What is ISAKMP?

ISAKMP stands for Internet Security

Association

Components of an IPSEC VPN that must be in place in order for it to function

the public traffic that is being forwarded between the client and VPN server or VPN server to VPN

server.

What are ESP and AH?

No, ESP is not Extra-Sensory
Percep
stands for Authen

Encapsula

ESP gives prot area indica wherea protected
data packet has been signed for integrity, and an Encrypted area
which indicates the informa
tunneled, ESP protects only the IP data payload (hence the name), and not the IP header.

ESP may be used to ensure nonrepudiation -replay service (a form of p
sequence integrity which guards against the use of commands or
credentials captured through password sniffing or similar a

Authen

Authen protocol suite, which authen of IP packets (datagrams) and guarantees
integrity of the data. The AH confirms the the
contents (both the header and payload) have not been changed since transmission. If
security been established, AH can
be
a
down technique.4

How Do They All Work Together?

When properly configured, an IPSEC VPN provides
mu
security mode and integrity of the data that is

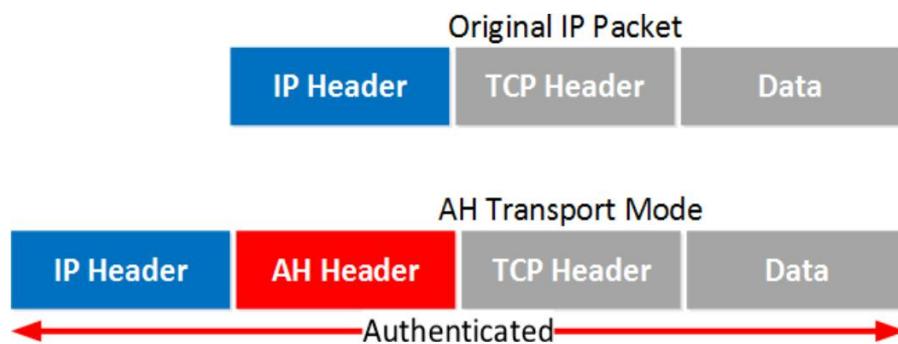
being transmitted. The data has not been intercepted and altered in transit and that they can rely on what they are seeing.

IPsec Protocols

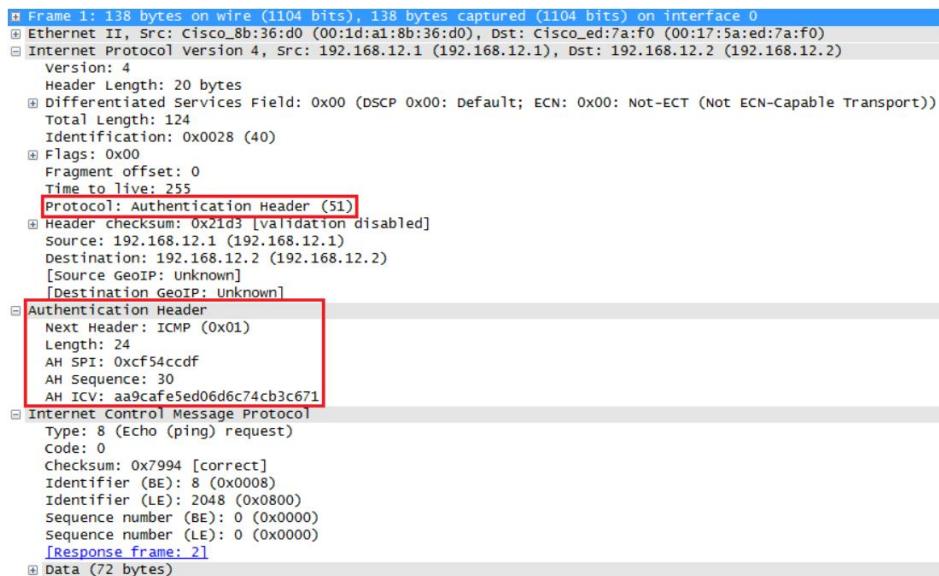
AH and/or ESP are the two protocols that we use to actually protect user data. Both of them can be used in transport or tunnel mode, let's walk through all the possible Authen Protocol

AH offers authentication by calculating a checksum that can be changed to encrypt the data.

in transit (TTL and header checksum). Let's start with transport mode... Transport mode is simple, it just adds an AH header after the IP header. Here's an example of an IP packet that carries some TCP traffic:



And here's what that looks like in Wireshark:



```
Frame 1: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on interface 0
Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a8:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 124
    Identification: 0x0028 (40)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Authentication Header (51)
    Header checksum: 0x21d3 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
        [Source GeoIP: Unknown]
        [Destination GeoIP: Unknown]
    Authentication Header
        Next Header: ICMP (0x01)
        Length: 24
        AH SPI: 0xcf54ccdf
        AH Sequence: 30
        AH ICV: aa9cafe5ed06d6c74cb3c671
    Internet Control Message Protocol
        Type: 8 (Echo (ping) request)
        Code: 0
        Checksum: 0x7994 [correct]
        Identifier (BE): 8 (0x0008)
        Identifier (LE): 2048 (0x0800)
        Sequence number (BE): 0 (0x0000)
        Sequence number (LE): 0 (0x0000)
        [Response frame: 2]
    Data (72 bytes)
```

Above you can see the AH header in between the IP header and ICMP header. This is a capture I

took of a ping between two routers. You can see that AH uses 5 fields:
Next Header: this iden

Length: this is the length of the AH header.

SPI (Security Parameters Index): this is an 32-bit identifier that the packet belongs.

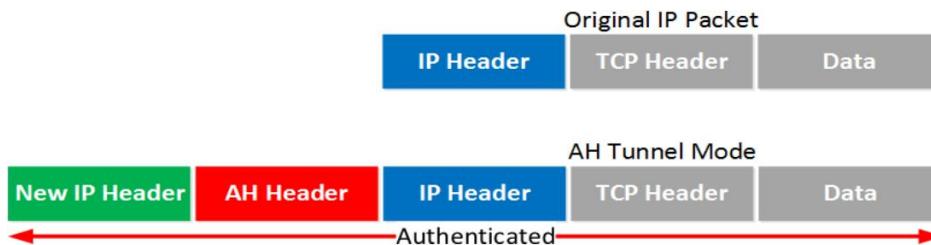
Sequence: this is the sequence number that helps against replay attacks.

ICV (Integrity Check Value): this is the calculated hash for the entire packet. It calculates a hash, when it's not the same you know something is wrong.

Let's c

could be useful when you are using private IP addresses and you need to tunnel your traffic over

the Internet. It's possible with AH but it doesn't offer encryption:



The en

```
Frame 1: 158 bytes on wire (1264 bits), 158 bytes captured (1264 bits) on interface 0
Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 144
    Identification: 0x0215 (533)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Authentication Header (51)
    Header checksum: 0x1fd2 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: unknown]
    [Destination GeoIP: unknown]
    Authentication Header
        Next Header: IPIP (0x04)
        Length: 24
        AH SPI: 0x646adc80
        AH Sequence: 5
        AH ICV: 606d214066853c0390cfef577
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 100
    Identification: 0x003c (60)
    Flags: 0x00
        0... .... = Reserved bit: Not set
        .0... .... = Don't fragment: Not set
        ..0.... = More fragments: Not set
    Fragment offset: 0
    Time to live: 255
    Protocol: ICMP (1)
    Header checksum: 0x2209 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: unknown]
    [Destination GeoIP: unknown]
    Internet Control Message Protocol
```

Above you can see the new IP header, then the AH header and finally the original IP packet that carries some ICMP traffic.

One problem with AH is that it doesn't play well with NAT / PAT. Fields in the IP header like TTL and the checksum are excluded by AH because it knows these will change. The IP addresses and port numbers however are included. If you change these with NAT, the ICV of AH fails.

Let's c

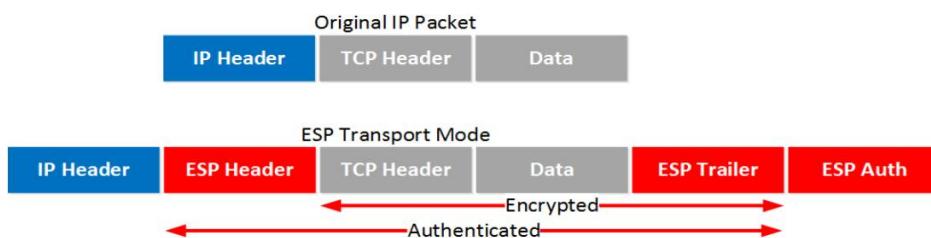
ESP

(Encapsula

ESP is the more popular choice of the two since it allows you to encrypt IP traffic. We can use it in transport or tunnel mode, let's look at both.

Transport Mode

When we use transport mode, we use the original IP header and insert an ESP header. Here's what it looks like:



Above you can see that we add an ESP header and trailer. Our transport layer (TCP for example) payload will be encrypted. It also offers authentication but unlike AH, it's not for the entire IP packet.

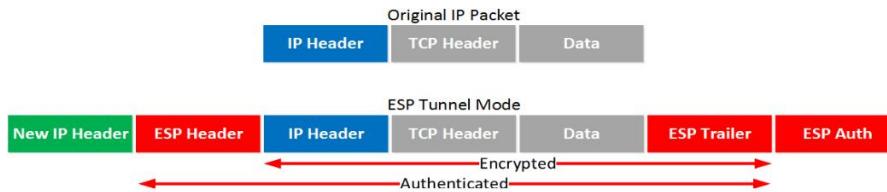
Here's what it looks like in wireshark:

```
# Frame 1: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits) on interface 0
# Ethernet II, Src: Cisco_B8:36:d0 (00:1d:a1:b8:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
# Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
# Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 152
    Identification: 0x0042 (66)
    Flags: 0x00
        0... .... = Reserved bit: Not set
        .0. .... = Don't fragment: Not set
        ..0. .... = More fragments: Not set
    Fragment offset: 0
    Time to live: 255
# Protocol: Encap Security Payload (50)
# Header checksum: 0x219e [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
# Encapsulating security payload
    ESP SPI: 0x36cb42df (919290591)
    ESP Sequence: 1
```

Above you can see the original IP packet and that we are using ESP. The IP header is in cleartext but everything else is encrypted.

Tunnel Mode

How about ESP in tunnel mode? This is where we use a new IP header which is useful for site-to-site VPNs:



It's similar to transport mode but we add a new header. The original IP header is now also encrypted.

Here's what it looks like in wireshark:

```
Frame 2: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits) on interface 0
Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
        Total Length: 168
        Identification: 0x023a (574)
        Flags: 0x00
        Fragment offset: 0
        Time to live: 255
    Protocol: Encap Security Payload (50)
    Header checksum: 0xf92 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
        [Source GeoIP: Unknown]
        [Destination GeoIP: unknown]
Encapsulating Security Payload
ESP SPI: 0xb8b181a7 (2343666087)
ESP Sequence: 5
```

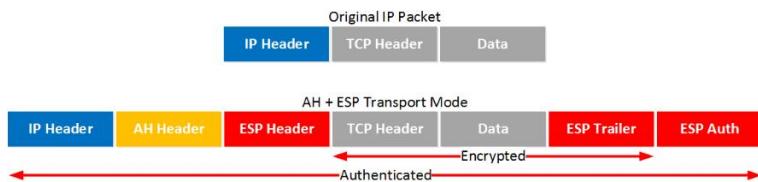
The output of the capture is above is similar to what you have seen in transport mode. The only difference is that this is a new IP header, you don't get to see the original IP header.

AH and ESP

This one confuses a lot of people, it's possible to use AH and ESP at the same out!

Transport Mode

Let's start with transport mode, here's what the IP packet will look like:



With transport mode we will use the original IP header, followed by an AH and ESP header. The transport layer, payload and ESP trailer will be encrypted.

Because we also use AH, the IP packet is authen en

Here's what it looks like in wireshark:

Above you can see the original IP packet, the AH header and the ESP header.

Conclusion:

Hence we had studied the IPsec (ESP and AH) protocol by capturing the packets using Wireshark tool.

ipsec.cap [Wireshark 1.8.0 (SVN Rev 43431 from /trunk-1.8)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.140.205	192.168.140.200	ISAKMP	294	Identity Protection (Main Mode)
2	0.014556	192.168.140.200	192.168.140.205	ISAKMP	214	Identity Protection (Main Mode)
3	0.042441	192.168.140.205	192.168.140.200	ISAKMP	82	Informational
4	10.054177	192.168.140.200	192.168.140.205	ISAKMP	214	Identity Protection (Main Mode)
5	10.073018	192.168.140.205	192.168.140.200	ISAKMP	270	Identity Protection (Main Mode)
6	10.090702	192.168.140.200	192.168.140.205	ISAKMP	270	Identity Protection (Main Mode)
7	10.104128	192.168.140.205	192.168.140.200	ISAKMP	110	Identity Protection (Main Mode)
8	10.105329	192.168.140.200	192.168.140.205	ISAKMP	110	Identity Protection (Main Mode)
9	10.108102	192.168.140.205	192.168.140.200	ISAKMP	198	Quick Mode
10	10.109646	192.168.140.200	192.168.140.205	ISAKMP	198	Quick Mode
11	10.109816	192.168.140.205	192.168.140.200	ISAKMP	94	Quick Mode

Frame 9: 198 bytes on wire (1584 bits), 198 bytes captured (1584 bits)
 Ethernet II, Src: VMware_c5:7d:db (00:0c:29:c5:7d:db), Dst: VMware_4f:ee:a2 (00:0c:29:4f:ee:a2)
 Internet Protocol Version 4, Src: 192.168.140.205 (192.168.140.205), Dst: 192.168.140.200 (192.168.140.200)
 User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
 Internet Security Association and Key Management Protocol
 Initiator cookie: c6d1459285150c7e
 Responder cookie: 6e961f01bf179b35
 Next payload: Hash (8)
 Version: 1.0
 Exchange type: Quick Mode (32)
 Flags: 0x01
 Message ID: 0x221faf3c
 Length: 156
 Encrypted data (128 bytes)

Encrypted Data (isakmp.enc.data), 128 bytes

capture.pcap - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: Expression... Clear Apply

No.	Len	Time	Source	Destination	Protocol	Info
1	198	2006-02-23 07:14:16.132126	192.168.0.3	10.0.0.2	ICMP	Echo (ping) request
2	226	2006-02-23 07:14:16.132135	10.0.0.1	10.0.0.2	ICMP	Destination unreachable
3	198	2006-02-23 07:14:17.132008	192.168.0.3	10.0.0.2	ICMP	Echo (ping) request
4	226	2006-02-23 07:14:17.132096	10.0.0.2	10.0.0.1	ICMP	Destination unreachable
5	198	2006-02-23 07:14:18.133007	192.168.0.3	10.0.0.2	ICMP	Echo (ping) request
6	226	2006-02-23 07:14:18.133085	10.0.0.2	10.0.0.1	ICMP	Destination unreachable

Frame 1: 198 bytes on wire (1584 bits), 198 bytes captured (1584 bits)
 Ethernet II, Src: Asustek_C0:d9:5b (00:0e:a6:0d:d9:5b), Dst: Digitalis_81:e3:57 (00:0c:29:c5:7d:db)
 Internet Protocol, Src: 10.0.0.1 (10.0.0.1), Dst: 10.0.0.2 (10.0.0.2)
 Encapsulating Security Payload
 ESP SPI: 0x0000000a
 ESP Sequence: 240
 ESP IV: E035E3F9212BA7B1D0DE2A5DA6D6D0E3
 Pad
 ESP Pad Length: 6
 Next header: IP(0x45)
 Authentication Data [correct]
 Internet Protocol, Src: 192.168.0.3 (192.168.0.3), Dst: 10.0.0.2 (10.0.0.2)
 Encapsulating Security Payload
 ESP SPI: 0x0000000f

Frame (198 bytes) Decrypted Data (140 bytes) Decrypted Data (84 bytes)

File: C:\cygwin\tmp\t2\capture.pcap" 153... | Packets: 782 Displayed: 782 Marked: 0 Time: 00:00:00.000 | Profile: Default

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Standard Input - IPsec GDOI to BPF (0/D)

No.	Time	Source	Destination	Protocol	Length	Info
90	62.805852	8.8.11.2	8.8.10.2	ESP	150	ESP (SPI=0x9ddaa3337)
91	63.803898	8.8.10.2	8.8.11.2	ESP	150	ESP (SPI=0xb89c0d01)
92	63.812929	8.8.11.2	8.8.10.2	ESP	150	ESP (SPI=0x9ddaa3337)
95	64.803473	8.8.10.2	8.8.11.2	ESP	150	ESP (SPI=0xb89c0d01)
96	64.808674	8.8.11.2	8.8.10.2	ESP	150	ESP (SPI=0x9ddaa3337)
99	65.804774	8.8.10.2	8.8.11.2	ESP	150	ESP (SPI=0xb89c0d01)
1	65.809123	8.8.11.2	8.8.10.2	ESP	150	ESP (SPI=0x9ddaa3337)

Frame 99: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits) on interface 0
 Ethernet II, Src: 00:35:5d:4b:5a:01 (00:35:5d:4b:5a:01), Dst: 00:35:5d:aa:04:00 (00:35:5d:aa:04:00)
 Internet Protocol Version 4, Src: 8.8.10.2, Dst: 8.8.11.2
 0.0.0.0... Version: 4
 0.0.0.1 = Header Length: 20 bytes (5)
 Differentiated Services Field: 0x000 (DSCP: CS0, ECN: Not-ECT)
 Total Length: 130
 Identification: 0x00d8 (216)
 Flags: 0x02 (Don't Fragment)
 Fragment offset: 0
 Time to live: 254
 Protocol: Encap Security Payload (50)
 Header checksum: 0x5658 [Validation disabled]
 [Header checksum status: Unverified]
 Source: 8.8.10.2
 Destination: 8.8.11.2
 [Source GeoIP: Unknown]

No. Time Source Destination Protocol Length Info

7.7.948776.. 12.0.0.1 23.0.0.1 ESP 162 ESP (SPI=0xdclf45c1)[Malformed Packet]

8.10.00340.. aa:bb:cc:00:00.. aa:bb:cc:00:00.. 162 ESP (SPI=0x70fc225e)

10.12.37226.. 12.0.0.1 23.0.0.1

Mark/Unmark Packet(s) Ctrl+M
 Ignore/Unignore Packet(s) Ctrl+D
 Set/Inset Time Reference Ctrl+T
 Time Shift... Ctrl+Shift+T
 Packet Comment... Ctrl+Alt+C
 Edit Resolved Name
 Apply as Filter
 Prepare as Filter
 Conversation Filter
 Colorize Conversation
 SCTP
 Follow
 Copy
 Decode As...
 Show Packet in New Window
 Protocol Preferences
 Open Encapsulating Security Payload preferences...
 Attempt to detect/decode NULL encrypted ESP payloads
 Check sequence numbers of ESP frames
 Attempt to detect/decode encrypted ESP payloads
 Attempt to Check ESP Authentication
 ESP SAs...
 Disable ESP...

0000 aa bb cc 00 02 00 aa bb cc 00 01 00 00 00 04 05 c9 E:
 0010 00 94 03 07 00 00 ff 32 00 6f 0c 00 01 00 00 00 04 05 00 E: 0
 0020 00 01 dc 0f 45 01 00 00 00 fd 45 c0 00 01 03 E: T:
 0030 00 00 ff 2f 96 b6 0c 00 00 17 00 01 00 00 00 00 00 00 E:
 0040 00 00 45 c0 00 3c 01 fa 00 00 01 38 29 95 ac 10 X:
 0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E:
 0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E:
 0070 01 00 00 00 00 00 00 00 00 00 00 00 17 00 02 00 00 E:
 0080 02 04 00 df d4 f6 b4 c2 0d 6f f1 a9 c9 3a 70 b: o: p:
 0090 40 0c 0b 30 00 41 af f3 eb 5c 6c 32 f0 b9 92 7f @: 0 A: \l: /:
 00a0 c5 1c ..