CNT4406/5412 Network Security

IPsec

Zhi Wang

Florida State University

Fall 2014

Introduction

IPsec is a protocol suite for securing IP communication by authenticating and encrypting each IP packet of a session.

Introduction

IPsec is a protocol suite for securing IP communication by authenticating and encrypting each IP packet of a session.

➠ IPsec can provide authentication and/or confidentiality

Introduction

IPsec is a protocol suite for securing IP communication by authenticating and encrypting each IP packet of a session.

- ➠ IPsec can provide authentication and/or confidentiality
- ➠ IPsec is implemented in the kernel, applications may remain unchanged

Introduction

IPsec is a protocol suite for securing IP communication by authenticating and encrypting each IP packet of a session.

- IPsec can provide authentication and/or confidentiality
- IPsec is implemented in the kernel, applications may remain unchanged
- IPsec can be configured to be transparent to users

Introduction...

Why do we need IPsec?

Introduction...

Why do we need IPsec?

- IPv4 has no authentication or integrity protection

Introduction...

Why do we need IPsec?

- IPv4 has no authentication or integrity protection
  ➠ IP spoofing, payload modification, lack of accountability

Introduction...

Why do we need IPsec?

- IPv4 has no authentication or integrity protection
  ➠ IP spoofing, payload modification, lack of accountability
- IPv4 has no confidentiality protection

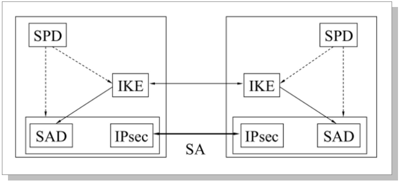
Introduction...

Why do we need IPsec?

- IPv4 has no authentication or integrity protection
  ➠ IP spoofing, payload modification, lack of accountability
- IPv4 has no confidentiality protection
  ➠ eavesdropping

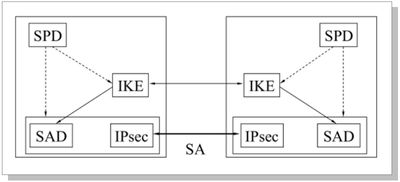
IPsec Architecture

- SPD: Security Policy Database



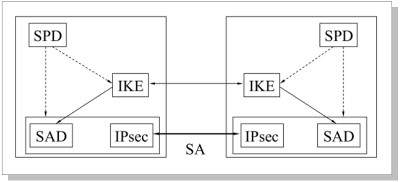
IPsec Architecture

- SPD: Security Policy Database
- IKE: Internet Key Exchange ➠ to negotiate security parameters



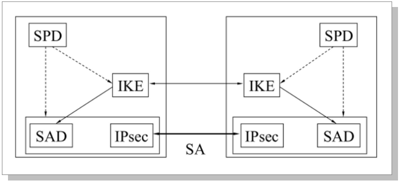
IPsec Architecture

- SPD: Security Policy Database
- IKE: Internet Key Exchange ➠ to negotiate security parameters
- SA & SAD: Security Association (Database)



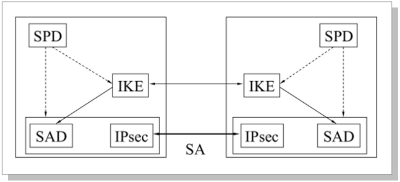
IPsec Architecture

- SPD: Security Policy Database
- IKE: Internet Key Exchange ➠ to negotiate security parameters
- SA & SAD: Security Association (Database)
- IPsec: Authentication Header/Encapsulating Security Payload
  ➠ AH $\rightarrow$ authentication
  ➠ ESP $\rightarrow$ encryption and/or authentication



Security Association

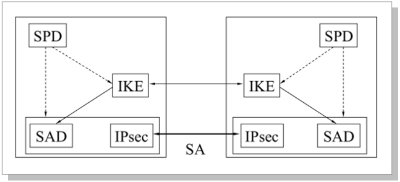
A IPsec **security association** is a cryptographically protected connection



Security Association

A IPsec **security association** is a cryptographically protected connection

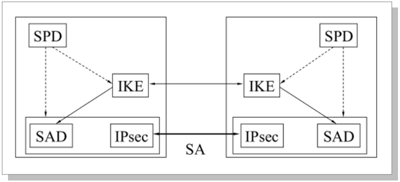
- A SA has a set of security parameters (attributes)
  - ➠ e.g., identities, algorithms, keys, sequence number
  - ➠ SA specifies how to process IPsec packets



Security Association

A IPsec **security association** is a cryptographically protected connection

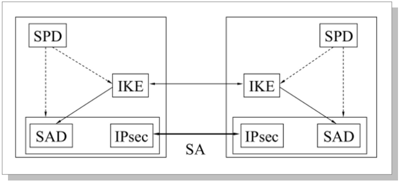
- A SA has a set of security parameters (attributes)
  - ➠ e.g., identities, algorithms, keys, sequence number
  - ➠ SA specifies how to process IPsec packets
- SA is unidirectional, two SAs for a conversation



Security Association

A IPsec **security association** is a cryptographically protected connection

- A SA has a set of security parameters (attributes)
  - ➠ e.g., identities, algorithms, keys, sequence number
  - ➠ SA specifies how to process IPsec packets
- SA is unidirectional, two SAs for a conversation
- SA may be changed during the conversation (IKE rekeying)



Security Parameter Index (SPI)

SPI is a 32-bit number assigned to a SA

Security Parameter Index (SPI)

SPI is a 32-bit number assigned to a SA

- SPI is chosen by the destination of a SA

Security Parameter Index (SPI)

SPI is a 32-bit number assigned to a SA

- SPI is chosen by the destination of a SA
- A SA is uniquely identified by $< SPI, destination\ addr, AH\ or\ ESP >$
  - ➠ SPI may overlap for unicast and multicast addresses
  - ➠ SPI may overlap for AH and ESP

Security Parameter Index (SPI)

SPI is a 32-bit number assigned to a SA

- SPI is chosen by the destination of a SA
- A SA is uniquely identified by $< SPI, destination\ addr, AH\ or\ ESP >$
  - ➠ SPI may overlap for unicast and multicast addresses
  - ➠ SPI may overlap for AH and ESP
- SPI is carried in each AH and ESP header
  - ➠ the receiver can look up the SA for the packet in its SAD
  - ➠ the SA determines how to process the packet

Security Association Database

- SAs are stored in the security association database

Security Association Database

- SAs are stored in the security association database
- SAD can be searched with $< SPI, destination\ addr, AH\ or\ ESP >$
  ➠ SA specifies how to send packets or process received packets

Security Association Database

- SAs are stored in the security association database
- SAD can be searched with $< SPI, destination\ addr, AH\ or\ ESP >$
  ➠ SA specifies how to send packets or process received packets
- Each host/gateway participating in IPsec maintains its own SAD

Security Policy Database

SPD is a database of policies to process packets: drop, send w/ IPsec? ...

Security Policy Database

SPD is a database of policies to process packets: drop, send w/ IPsec? ...

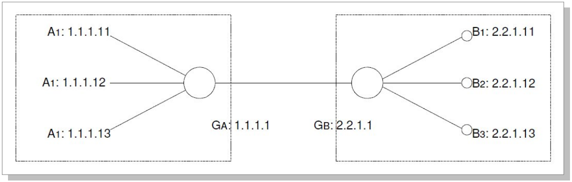
➠ SPD policy has a selector and action, similar to the firewall policy

Security Policy Database

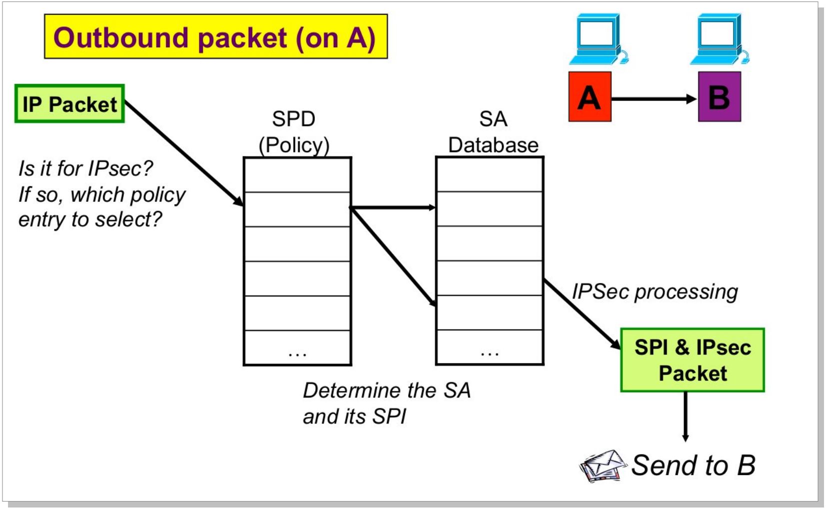
SPD is a database of policies to process packets: drop, send w/ IPsec? ...

➠ SPD policy has a selector and action, similar to the firewall policy

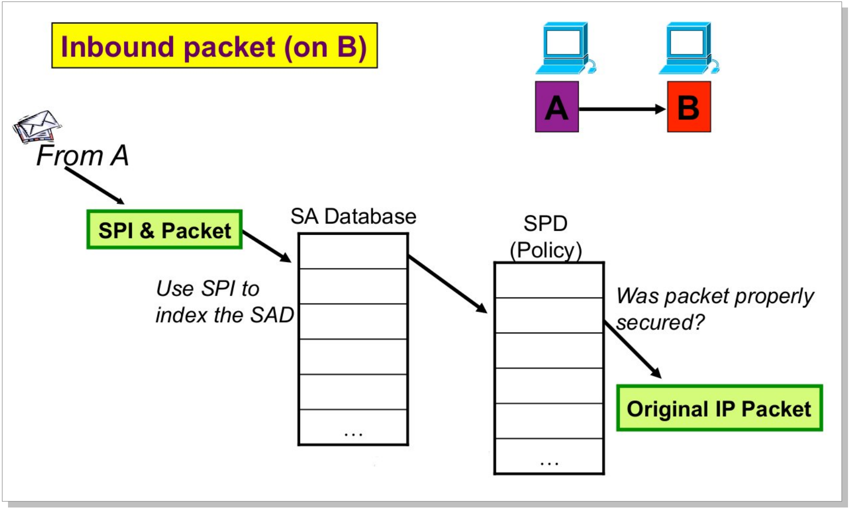
Index	Local	Remote	Proto	Dir	Action	SA In/Out
9	1.1.1.12:80	2.2.1.0/24:any	TCP	I/O	IPsec	sa15/sa25
8	1.1.1.0/24:any	2.2.1.0/24:any	any	I/O	IPsec	sa10/sa20
			...			
...	...			I		
...	...			O	drop	null
0	any	any	any	I/O	bypass	null



IPsec Outbound Processing

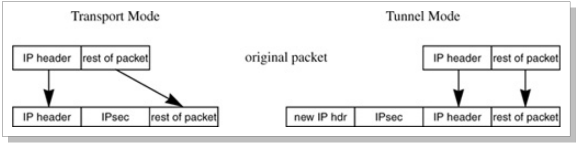


IPsec Inbound Processing



Tunnel Mode

IPsec can operate in **tunnel mode** and **transport mode**

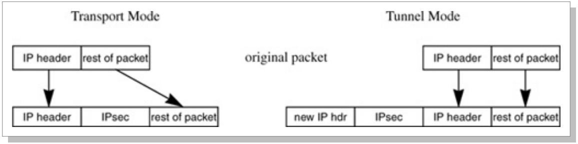


Tunnel Mode

IPsec can operate in **tunnel mode** and **transport mode**

Tunnel mode:

- the original IP packet is enclosed in an outer IP header w/ ESP/AH

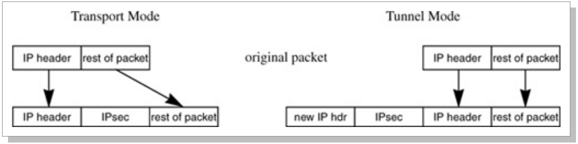


Tunnel Mode

IPsec can operate in **tunnel mode** and **transport mode**

Tunnel mode:

- the original IP packet is enclosed in an outer IP header w/ ESP/AH
- commonly used in firewall to firewall or endnode to firewall

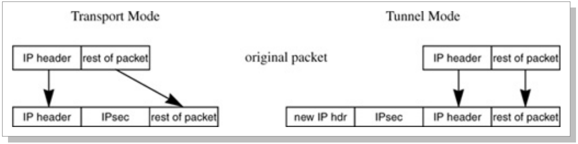


Tunnel Mode

IPsec can operate in **tunnel mode** and **transport mode**

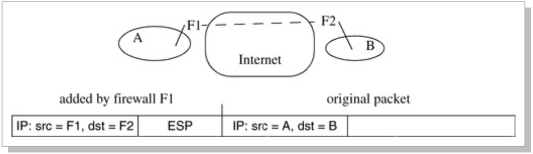
Tunnel mode:

- the original IP packet is enclosed in an outer IP header w/ ESP/AH
- commonly used in firewall to firewall or endnode to firewall
  ➠ data is only protected inside the tunnel (not end-to-end)



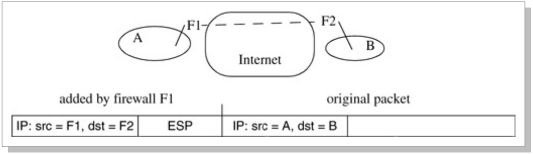
Tunnel Mode...

- Two firewalls establish an encrypted tunnel across the Internet



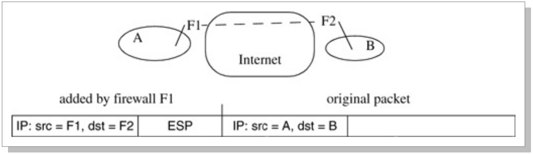
Tunnel Mode...

- Two firewalls establish an encrypted tunnel across the Internet
- IPsec packets from F_1 to F_2 have a destination of F_2



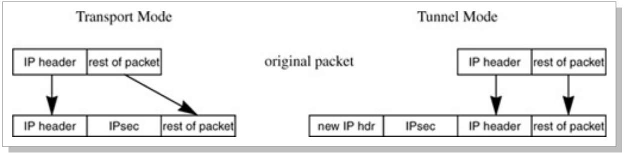
Tunnel Mode...

- Two firewalls establish an encrypted tunnel across the Internet
- IPsec packets from F_1 to F_2 have a destination of F_2
  ➠ the inner IP packet is not changed



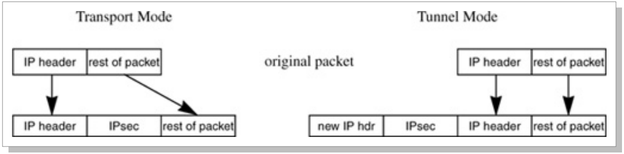
Transport Mode

- IPsec header is inserted after the IP header of the original packet



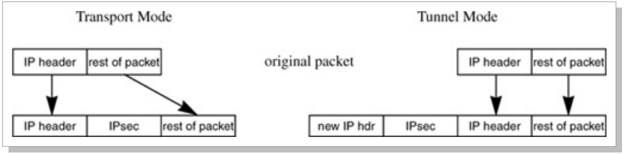
Transport Mode

- IPsec header is inserted after the IP header of the original packet
- Commonly applied end-to-end, data is protected end-to-end



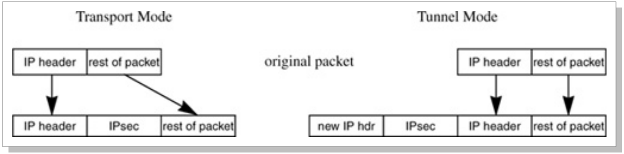
Transport Mode

- IPsec header is inserted after the IP header of the original packet
- Commonly applied end-to-end, data is protected end-to-end
- Transport mode is not strictly necessary, tunnel mode can be used



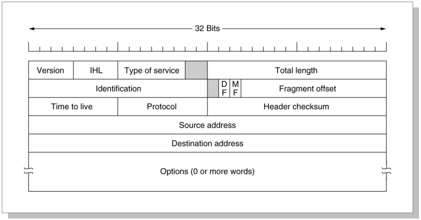
Transport Mode

- IPsec header is inserted after the IP header of the original packet
- Commonly applied end-to-end, data is protected end-to-end
- Transport mode is not strictly necessary, tunnel mode can be used
  ➠ tunnel mode uses more header space



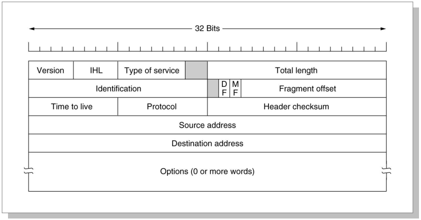
IP Header

- Protocol: what protocol follows the IP header



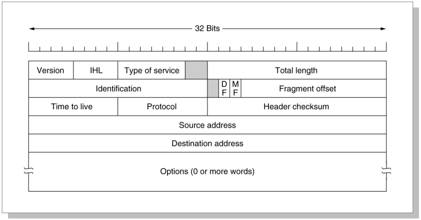
IP Header

- Protocol: what protocol follows the IP header
  ➠ common protocols: TCP(6), UDP(17), IP(4), ESP(50), AH(51)



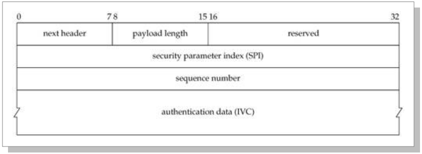
IP Header

- Protocol: what protocol follows the IP header
  - ➠ common protocols: TCP(6), UDP(17), IP(4), ESP(50), AH(51)
  - ➠ protocol headers in IPv6 are TLV-encoded



Authentication Header

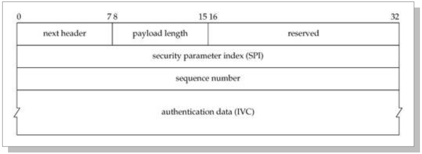
AH header provides authentication only, not encryption



Authentication Header

AH header provides authentication only, not encryption

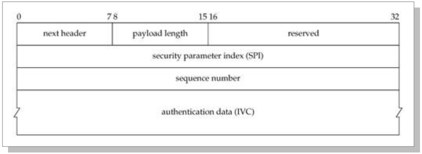
- AH header has variable length (which field?)
  - sequence number: sequence number of AH packets
  - ICV: integrity check value for the data



Authentication Header

AH header provides authentication only, not encryption

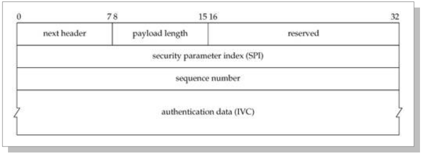
- AH header has variable length (which field?)
  - sequence number: sequence number of AH packets
  - ICV: integrity check value for the data
- ICV covers both data and immutable or predictable fields in IP header



Authentication Header

AH header provides authentication only, not encryption

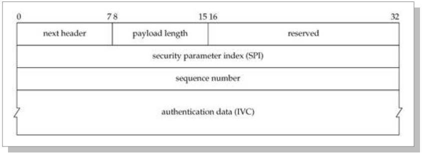
- AH header has variable length (which field?)
  - sequence number: sequence number of AH packets
  - ICV: integrity check value for the data
- ICV covers both data and immutable or predictable fields in IP header
  - outer IP header is covered in tunnel mode



Authentication Header

AH header provides authentication only, not encryption

- AH header has variable length (which field?)
  - sequence number: sequence number of AH packets
  - ICV: integrity check value for the data
- ICV covers both data and immutable or predictable fields in IP header
  - outer IP header is covered in tunnel mode
  - immutable fields: version, total length (what if fragmented?)...



AH Problems

- AH covers both data and part of the IP header, problems?

AH Problems

- AH covers both data and part of the IP header, problems?
  ➠ difficult for NAT traversal because NAT need to change IP header

AH Problems

- AH covers both data and part of the IP header, problems?
  - ➠ difficult for NAT traversal because NAT need to change IP header
  - ➠ inconsistent in which fields to include: fragment offset is mutable?

AH Problems

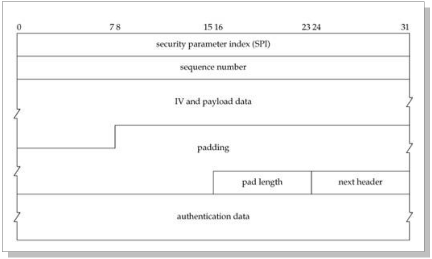
- AH covers both data and part of the IP header, problems?
  ➠ difficult for NAT traversal because NAT need to change IP header
  ➠ inconsistent in which fields to include: fragment offset is mutable?
- ICV before data prevents streamlining AH head process
  ➠ NIC needs to cache the whole packet, cannot send-as-you-go

AH Problems

- AH covers both data and part of the IP header, problems?
  ➠ difficult for NAT traversal because NAT need to change IP header
  ➠ inconsistent in which fields to include: fragment offset is mutable?
- ICV before data prevents streamlining AH head process
  ➠ NIC needs to cache the whole packet, cannot send-as-you-go
- AH can only do authentication and it duplicates functionality in ESP

Encapsulating Security Payload

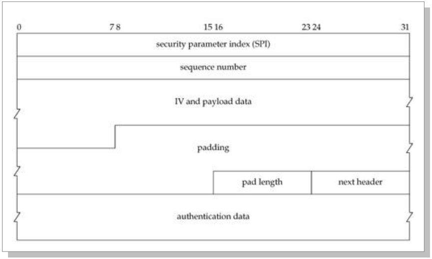
ESP always has an encryption operation, and also supports authentication



Encapsulating Security Payload

ESP always has an encryption operation, and also supports authentication

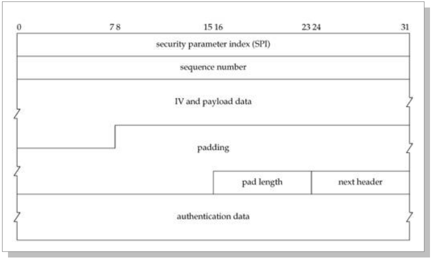
➠ special null encryption if encryption is not needed



Encapsulating Security Payload

ESP always has an encryption operation, and also supports authentication
➠ special null encryption if encryption is not needed

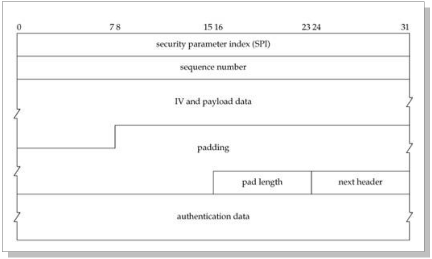
- ESP puts information before and after the data (sandwiching)
  ➠ ICV after the data avoids caching whole packet before sending it



Encapsulating Security Payload

ESP always has an encryption operation, and also supports authentication
➠ special null encryption if encryption is not needed

- ESP puts information before and after the data (sandwiching)
  ➠ ICV after the data avoids caching whole packet before sending it
- Data is padded to fit the cipher's block size



Summary

- IPsec Architecture
- IPsec Modes
- AH and ESP

- Next lecture: IPsec/IKE