CSC 474/574
Information Systems Security

Topic 7.2: IPsec

Outline

- IPsec Objectives
- IPsec architecture & concepts
- IPsec authentication header
- IPsec encapsulating security payload
IPsec Objectives

• Why do we need IPsec?
  – IP V4 has no authentication
    • IP spoofing
    • Payload could be changed without detection.
  – IP V4 has no confidentiality mechanism
    • Eavesdropping
  – Denial of service (DOS) attacks
    • Cannot hold the attacker accountable due to the lack of authentication.

IPsec Objectives (cont’d)

• IP layer security mechanism for IPv4 and IPv6
  – Not all applications need to be security aware
  – Can be transparent to users
  – Provide authentication and confidentiality mechanisms.
IPsec Architecture

SPD: Security Policy Database; IKE: Internet Key Exchange; SA: Security Association; SAD: Security Association Database.

IPsec Architecture (Cont’d)

- Two Protocols (Mechanisms)
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)
- IKE Protocol
  - Internet Key Management
  - Will be covered in CSC 774.
IPsec Architecture (Cont’d)

• Can be implemented in
  – Host or gateway
• Can work in two Modes
  – Tunnel mode
  – Transport mode

Hosts & Gateways

• Hosts can implement IPsec to connect to:
  – Other hosts in transport or tunnel mode
  – Or Gateways in tunnel mode
• Gateways to gateways
  – Tunnel mode
Tunnel Mode

Encrypted Tunnel

Gateway

A

Unencrypted

A

Encrypted

B

Unencrypted

Gateway

New IP Header

AH or ESP Header

Orig IP Header

TCP

Data

Tunnel Mode (Cont’d)

Outer IP header

IPsec header

Inner IP header

Higher layer protocol

Destination

IPsec entity

ESP

AH

Real IP destination

• ESP applies only to the tunneled packet
• AH can be applied to portions of the outer header
Transport Mode

![Diagram showing IP header, AH or ESP header, TCP, and data, with encrypted/authenticated data.]

Transport Mode (Cont’d)

- ESP protects higher layer payload only
- AH can protect IP headers as well as higher layer payload
Security Association (SA)

- An association between a sender and a receiver
  - Consists of a set of security related parameters
  - E.g., sequence number, encryption key
- One way relationship
- Determine IPsec processing for senders
- Determine IPsec decoding for destination
- SAs are not fixed! Generated and customized per traffic flows

Security Parameters Index (SPI)

- A bit string assigned to an SA.
- Carried in AH and ESP headers to enable the receiving system to select the SA under which the packet will be processed.
- 32 bits
- SPI + Dest IP address + IPsec Protocol
  - identifies each SA in SA Database (SAD)
SA Database (SAD)

- Holds parameters for each SA
  - Sequence number counter
  - Lifetime of this SA
  - AH and ESP information
  - Tunnel or transport mode

- Every host or gateway participating in IPsec has their own SA database

SA Bundle

- More than 1 SA can apply to a packet
- Example: ESP does not authenticate new IP header. How to authenticate?
  - Use SA to apply ESP w/out authentication to original packet
  - Use 2nd SA to apply AH
Security Policy Database (SPD)

- **Decide**
  - What traffic to protect?
  - Has incoming traffic been properly secured?
- **Policy entries define which SA or SA Bundles to use on IP traffic**
- **Each host or gateway has their own SPD**
- **Index into SPD by Selector fields**
  - Selectors: IP and upper-layer protocol field values.
  - Examples: Dest IP, Source IP, Transport Protocol, IPSec Protocol, Source & Dest Ports, …

SPD Entry Actions

- **Discard**
  - Do not let in or out
- **Bypass**
  - Outbound: do not apply IPSec
  - Inbound: do not expect IPSec
- **Protect** – **will point to an SA or SA bundle**
  - Outbound: apply security
  - Inbound: security must have been applied
SPD Protect Action

- If the SA does not exist…
  - Outbound processing
    - Trigger key management protocols to generate SA dynamically, or
    - Request manual specification, or
    - Other methods
  - Inbound processing
    - Drop packet

Outbound Processing

1. **Outbound packet (on A)**
   - **IP Packet**
   - **SPD (Policy)**
     - ... (Policy entries)
   - **SA Database**
     - ... (SA entries)

2. **Determine the SA and its SPI**

3. **IPSec processing**

4. **SPI & IPsec Packet**

5. **Send to B**

Is it for IPsec?
- If so, which policy entry to select?
Inbound Processing

Authentication Header (AH)

- Data integrity
  - Entire packet has not been tampered with
- Authentication
  - Can “trust” IP address source
  - Use MAC to authenticate
- Anti-replay feature
- Integrity check value
Integrity Check Value - ICV

- Message authentication code (MAC) calculated over
  - IP header fields that do not change or are predictable
  - IP header fields that are unpredictable are set to zero.
  - IPsec AH header with the ICV field set to zero.
  - Upper-level data
- Code may be truncated to first 96 bits

IPsec Authentication Header

```
SAD
```

```
<table>
<thead>
<tr>
<th>Next Header (TCP/UDP)</th>
<th>Payload Length 6-2=4</th>
<th>Reserved</th>
</tr>
</thead>
</table>
```

```
SPI
```

```
Sequence Number
```

```
ICV
```
Encapsulated Security Protocol (ESP)

- Confidentiality for upper layer protocol
- Partial traffic flow confidentiality (Tunnel mode only)
- Data origin authentication and connectionless integrity (optional)

Outbound Packet Processing

- Form ESP payload
- Pad as necessary
- Encrypt result [payload, padding, pad length, next header]
- Apply authentication
Outbound Packet Processing...

- Sequence number generation
  - Increment then use
  - With anti-replay enabled, check for rollover and send only if no rollover
  - With anti-replay disabled, still needs to increment and use but no rollover checking

- ICV calculation
  - ICV includes whole ESP packet except for authentication data field.
  - Implicit padding of ‘0’s between next header and authentication data is used to satisfy block size requirement for ICV algorithm
  - Not include the IP header.
Inbound Packet Processing

- Sequence number checking
  - Anti-replay is used only if authentication is selected
  - Sequence number should be the first ESP check on a packet upon looking up an SA
  - Duplicates are rejected!

<table>
<thead>
<tr>
<th>0</th>
<th>Sliding Window size $\geq 32$</th>
</tr>
</thead>
<tbody>
<tr>
<td>reject</td>
<td>Check bitmap, verify if new verify</td>
</tr>
</tbody>
</table>

Anti-replay Feature

- Optional
- Information to enforce held in SA entry
- Sequence number counter - 32 bit for outgoing IPsec packets
- Anti-replay window
  - 32-bit
  - Bit-map for detecting replayed packets
Anti-replay Sliding Window

- Window should not be advanced until the packet has been authenticated
- Without authentication, malicious packets with large sequence numbers can advance window unnecessarily
  - Valid packets would be dropped!

Inbound Packet Processing...

- Packet decryption
  - Decrypt quantity [ESP payload, padding, pad length, next header] per SA specification
  - Processing (stripping) padding per encryption algorithm; In case of default padding scheme, the padding field SHOULD be inspected
  - Reconstruct the original IP datagram
- Authentication verification (option)
ESP Processing - Header Location...

- Transport mode IPv4 and IPv6

**IPv4**

<table>
<thead>
<tr>
<th>Orig IP hdr</th>
<th>ESP hdr</th>
<th>TCP</th>
<th>Data</th>
<th>ESP trailer</th>
<th>ESP Auth</th>
</tr>
</thead>
</table>

**IPv6**

<table>
<thead>
<tr>
<th>Orig IP hdr</th>
<th>Orig ext hdr</th>
<th>ESP hdr</th>
<th>TCP</th>
<th>Data</th>
<th>ESP trailer</th>
<th>ESP Auth</th>
</tr>
</thead>
</table>

ESP Processing - Header Location...

- Tunnel mode IPv4 and IPv6

**IPv4**

<table>
<thead>
<tr>
<th>New IP hdr</th>
<th>ESP hdr</th>
<th>Orig IP hdr</th>
<th>TCP</th>
<th>Data</th>
<th>ESP trailer</th>
<th>ESP Auth</th>
</tr>
</thead>
</table>

**IPv6**

<table>
<thead>
<tr>
<th>New IP hdr</th>
<th>New ext hdr</th>
<th>ESP hdr</th>
<th>Orig IP hdr</th>
<th>Orig ext hdr</th>
<th>TCP</th>
<th>Data</th>
<th>ESP trailer</th>
<th>ESP Auth</th>
</tr>
</thead>
</table>