IPSec: Communication Security Within The Network

Antonio Carzaniga

Faculty of Informatics
Università della Svizzera italiana

May 15, 2020

Outline

- Introduction: security at different layers
- Basic concepts of internetworking
- Architecture of IPSec
- Security services: AH and ESP
- AH format
- AH outbound traffic processing
- AH inbound traffic processing
- ESP format
- ESP outbound traffic processing
- ESP inbound traffic processing
- Policies and security associations (SAs)
- Automated key and SA management

Main references

- RFC 4301: Security Architecture for the Internet Protocol (IPsec overview)
- RFC 4302: IP Authentication Header
- RFC 4303: IP Encapsulating Security Payload
- RFC 4306: Internet Key Exchange (IKEv2) Protocol

1. E-mail security
   - application-specific

1. E-mail security
   - application-specific

   Solution: *PGP, S/MIME*

1. E-mail security
   - application-specific

   Solution: *PGP, S/MIME*

2. E-commerce, secure WEB access
   - short-lived associations (e.g., HTTP sessions)
   - application-independent (HTTP, IMAP, SMTP)

1. E-mail security
   - application-specific

   Solution: *PGP, S/MIME*

2. E-commerce, secure WEB access
   - short-lived associations (e.g., HTTP sessions)
   - application-independent (HTTP, IMAP, SMTP)

   Solution: *TLS/SSL*

1. E-mail security
   - application-specific

   Solution: *PGP, S/MIME*

2. E-commerce, secure WEB access
   - short-lived associations (e.g., HTTP sessions)
   - application-independent (HTTP, IMAP, SMTP)

   Solution: *TLS/SSL*

3. Accessing services from the open Internet
   - short- and medium-term associations (work sessions)
   - more than one tool/protocol: e-mail, file access (e-mail, GIT, rsync), printing, etc.
   - still requires application-specific configuration

   Solution: *application- or system-level tunnels/proxies such as SSH*

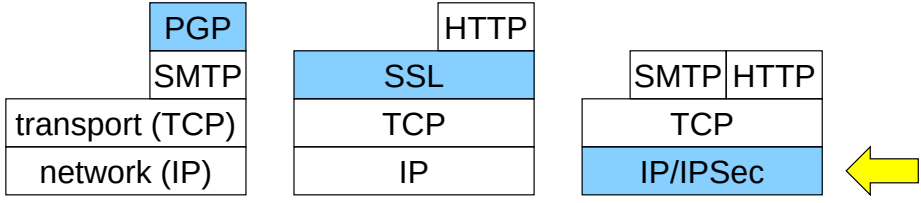
4. Branches of the same company, or research groups at different universities
   - long-term associations
   - VPNs
   - several applications and protocols (DB, old terminals, experimental distributed systems, etc.)

4. Branches of the same company, or research groups at different universities
   - long-term associations
   - VPNs
   - several applications and protocols (DB, old terminals, experimental distributed systems, etc.)

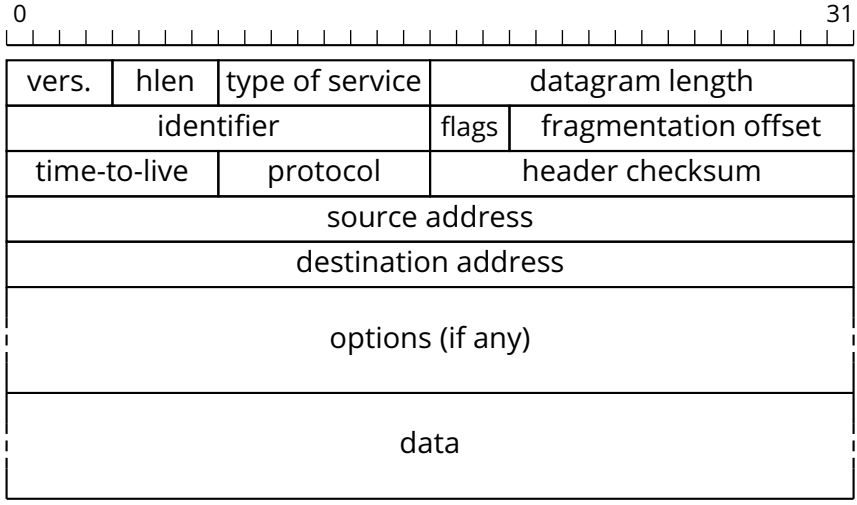
   Solution: *network-level security: IPSec*

4. Branches of the same company, or research groups at different universities
   - long-term associations
   - VPNs
   - several applications and protocols (DB, old terminals, experimental distributed systems, etc.)

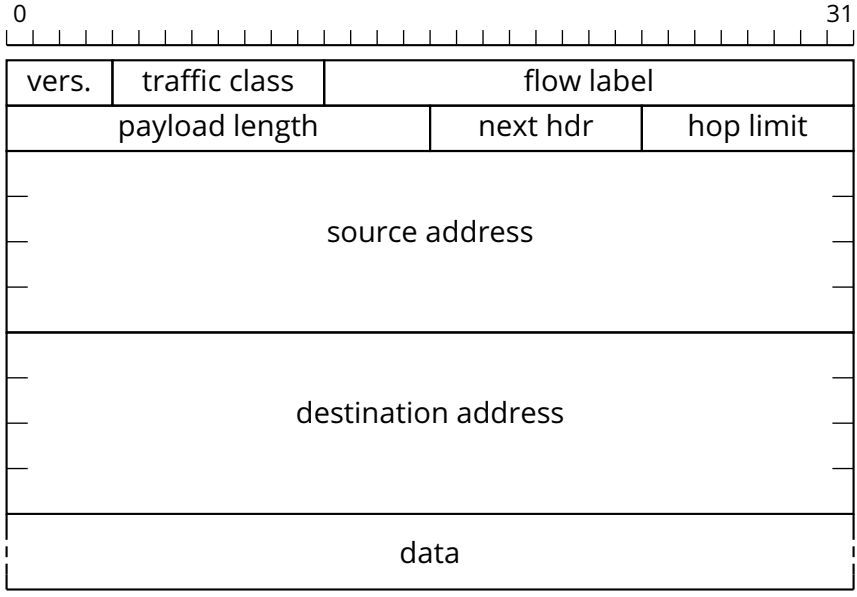
Solution: *network-level security: IPSec*



IPv4 Datagram Format



IPv6 Datagram Format



IP-Level Security: Goals

IPsec is designed to provide interoperable, high quality, cryptographically-based security for IPv4 and IPv6.
In particular:

- Access control
- Connectionless integrity
- Data origin authentication
- Protection against replays (a form of partial sequence integrity)
- Confidentiality (encryption)
- Limited traffic flow confidentiality.

These services are provided at the IP layer, offering protection for IP and/or upper layer protocols.

General Benefits of IPSec

- Network-, transport-, and application-level
  - virtual private networks
  - transparent security for applications and users
- Routing infrastructure
  - authentication for routing advertisements

How Does IPSec Apply Protection to Traffic?

- Applications are oblivious to IPSec
- So who decides where, when, and how to apply security?

How Does IPSec Apply Protection to Traffic?

- Applications are oblivious to IPSec
- So who decides where, when, and how to apply security?
- *For each packet*, an IPSec implementation decides whether to
  - discard that packet
  - bypass IPSec security services, or
  - afford IPSec security services

How Does IPSec Apply Protection to Traffic?

- Applications are oblivious to IPSec
- So who decides where, when, and how to apply security?
- *For each packet*, an IPSec implementation decides whether to
  - discard that packet
  - bypass IPSec security services, or
  - afford IPSec security services
- In other words, ***applying IPSec security services is largely a network management decision***

IPsec uses two protocols to provide traffic security:

IPsec uses two protocols to provide traffic security:

- ***Authentication Header (AH)*** provides connectionless integrity, data origin authentication, and an optional anti-replay service.

How IPSec Provides Security

IPsec uses two protocols to provide traffic security:

- ***Authentication Header (AH)*** provides connectionless integrity, data origin authentication, and an optional anti-replay service.
- ***Encapsulating Security Payload (ESP)*** provides
  - confidentiality (encryption), and limited traffic flow confidentiality, and optionally
  - data origin authentication, and an anti-replay service

IPsec uses two protocols to provide traffic security:

- ***Authentication Header (AH)*** provides connectionless integrity, data origin authentication, and an optional anti-replay service.
- ***Encapsulating Security Payload (ESP)*** provides
  - confidentiality (encryption), and limited traffic flow confidentiality, and optionally
  - data origin authentication, and an anti-replay service

Both AH and ESP, indirectly, provide access control, based on the distribution of cryptographic keys and the management of traffic flows relative to these security protocols.

Both AH and ESP can function in either

Both AH and ESP can function in either

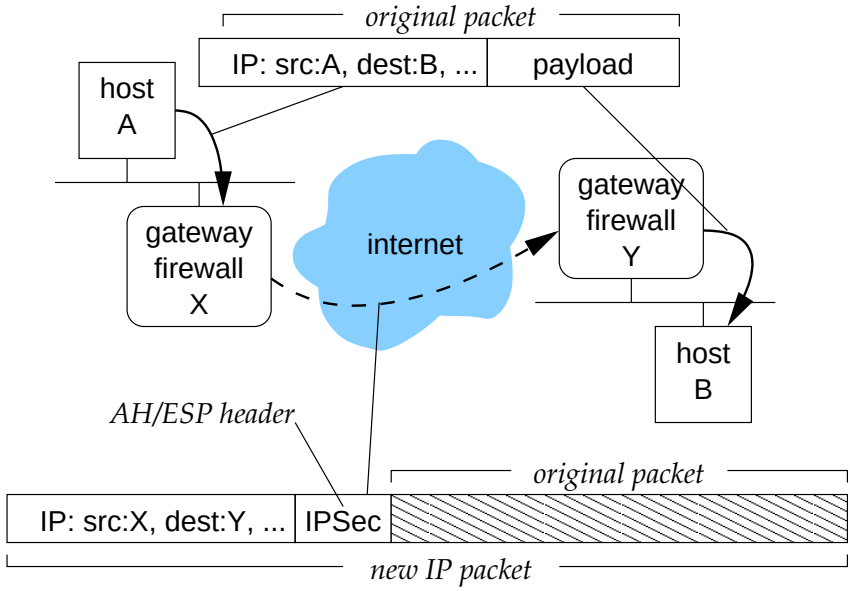
- ***transport mode*** provides security for upper-level protocols (such as TCP or UDP) by authenticating and/or encrypting the payload

Transport and Tunnel Modes

Both AH and ESP can function in either

- ***transport mode*** provides security for upper-level protocols (such as TCP or UDP) by authenticating and/or encrypting the payload
- ***tunnel mode*** provides security for the whole IP packet by encapsulating (tunneling) that packet into another IP packet.

Virtual Private Networks with Tunnel Mode



Security Associations (SA)

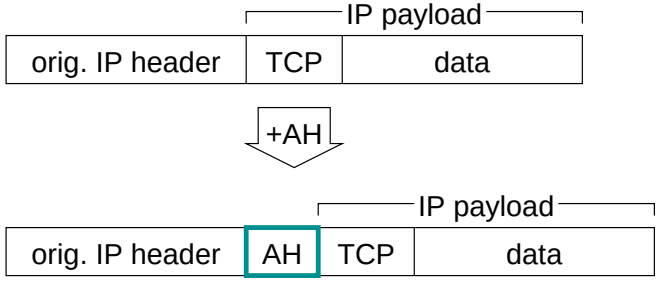
- Key concept appearing in both authentication and confidentiality mechanisms of IPSec
- ***One-way*** relation, between a *sender* and a *receiver*

Security Associations (SA)

- Key concept appearing in both authentication and confidentiality mechanisms of IPSec
- ***One-way*** relation, between a *sender* and a *receiver*
- A security association is identified by
  - ***security parameter index (SPI):*** a key that identifies a set of parameters for this association, stored in the *Security Association Database*
  - ***destination address*** (currently only unicast addresses)
  - ***security protocol identifier:*** AH or ESP

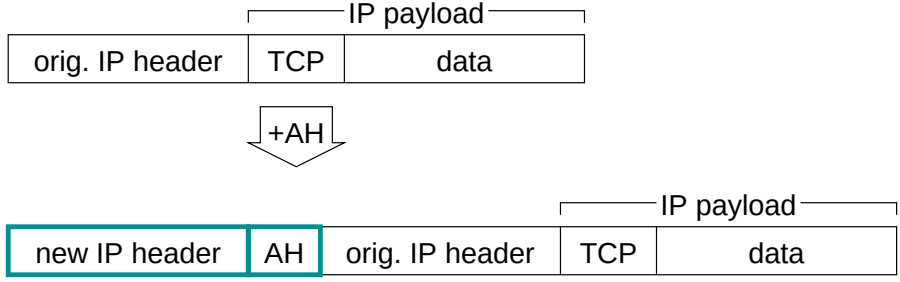
Transport Mode for AH

For *transport mode AH*, the authentication header is inserted after the original IP header and before the IP payload (with minor differences between IPv4 and IPv6):

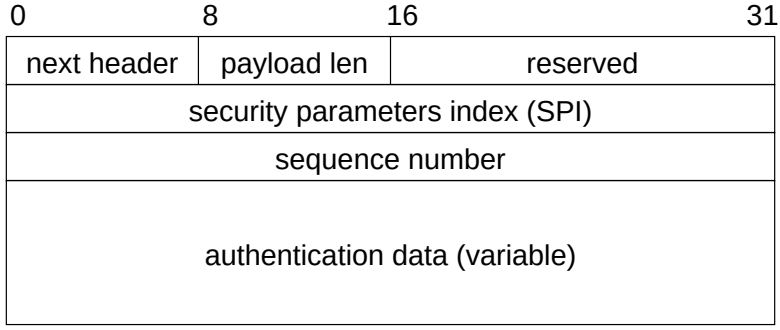


Tunnel Mode for AH

For *tunnel mode AH*, the original IP packet is encapsulated within another IP packet. The authentication header authenticates the entire original IP packet, and is inserted between the outer IP header and the inner (original) IP header.



Authentication Header Format



- ***payload length:*** length of this header in 32-bit words minus 2
  - common feature of all IPv6 extension headers
  - e.g., for a 96-bit authentication value (using HMAC-MD5-96 or HMAC-SHA1-96), *payload length* would be 4.
- ***reserved:*** a 16-bit field reserved for future use
  - must be set to 0 (included in the scope of the authentication function)

AH Outbound Processing

1. *Security association lookup*
2. *Sequence number generation*
   - sequence number is initialized to 0 when the SA is establised
   - if anti-replay service enabled: sequence number is incremented for each packet
   - if anti-replay service enabled: sender checks for sequence number overflow. Overflow is not permitted, so a new SA must be established.

AH Outbound Processing (2)

3. *Integrity code calculation*

Authentication data contains the MAC for the packet. It is a *variable-length* field. The current specification mentions two MAC algorithms:

- HMAC-MD5-96
- HMAC-SHA-1-96

The value of *authentication data* (MAC) is computed over

- ***immutable IP header fields***, for example:
  - IPv4 Internet Header Length
  - IPv6 Version
  - Source Address
- ***predictable IP header fields***, for example:
  - destination address
- ***AH header*** other than the Authentication Data field
- ***the entire payload***, which is immutable in transit

AH Outbound Processing (3)

3. *Integrity code calculation* (...continued)
   Mutable fields are zeroed when computing the MAC.

   For example, this is the classification of header fields for the IPv6 (base) header:

   - *Immutable*
     - Version
     - Payload Length
     - Next Header (this should be the value for AH)
     - Source Address
     - Destination Address (without Routing Extension Header)
   - *Mutable but predictable*
     - Destination Address (with Routing Extension Header)
   - *Mutable* (zeroed prior to ICV calculation)
     - Class
     - Flow Label
     - Hop Limit

AH Outbound Processing (4)

3. *Integrity code calculation* (...continued)

   - It might be necessary to add some ***padding*** to the data in order to compute the integrity code
     - Necessary padding must be added at the end of the packet
     - it must be *zero*
     - it is not transmitted.

   - ***fragmentation:*** if necessary, fragmentation occurs only *after* AH processing

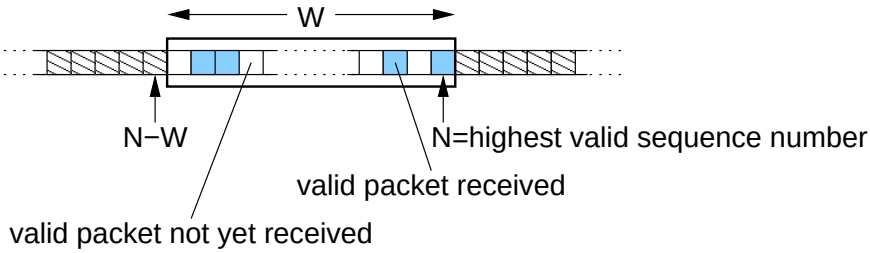
Inbound Traffic Processing

1. *Fragmentation reassembly:* fragments are reassembled as usual

Inbound Traffic Processing

1. *Fragmentation reassembly:* fragments are reassembled as usual
2. *Security association lookup*
    1. if SA (IP.dest, SPI, 'AH') is not in the SA database: ***drop packet***
    2. if SA has *anti-replay protection*: ***sequence number verification***
    3. otherwise: proceed with ***integrity check***

3. *Sequence number verification*



- Receiver must keep track of a fixed *window* of packets (fixed size W, at least $W = 32$, default $W = 64$)

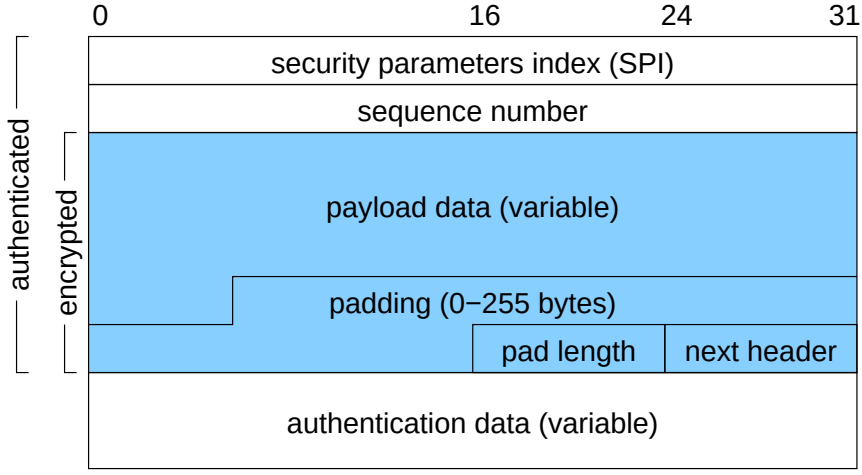
Inbound Traffic Processing (3)

4. *Integrity check*:
    1. save the ICV
    2. replace the bytes with zeroes
    3. zero out all other mutable fields
    4. add (zero) padding if necessary
    5. compute the MAC and compare with saved ICV

packets that fail the integrity verification must be discarded

Encapsulating Security Payload (ESP)

Encapsulating Security Payload (ESP)

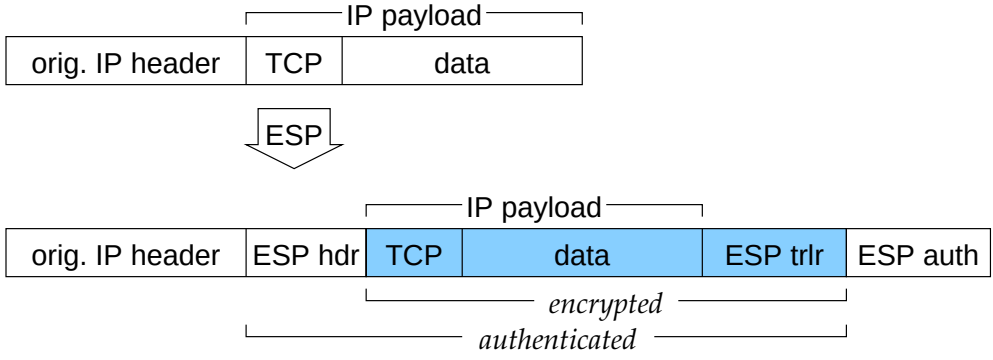


- Confidentiality of payload
- Integrity (i.e., authentication)
- Protection against replay attacks

Transport mode for ESP

Transport mode for ESP

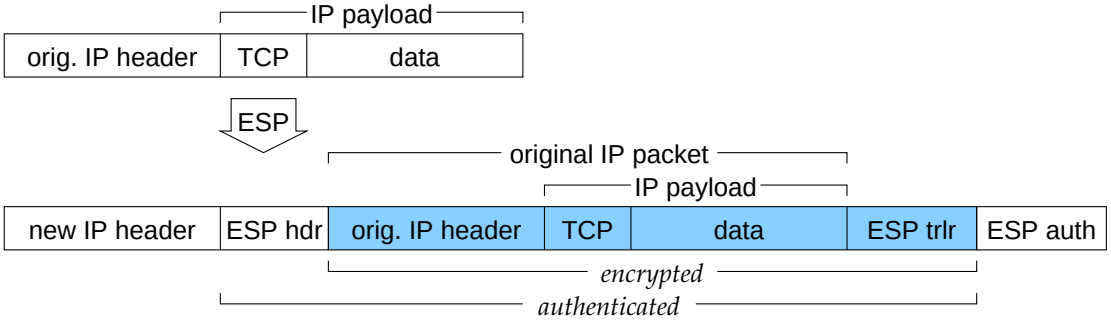
Transport mode ESP can be seen as a simple IP payload: the ESP packet is inserted after the original IP header (with some minor differences between IPv4 and IPv6):



Tunnel mode for ESP

Tunnel mode for ESP

For *tunnel mode ESP*, the original IP packet is encapsulated within another IP packet as an encrypted and (optionally) authenticated ESP:



Authentication and Reply Protection in ESP

- *Protection against replay attack:* same as in AH
  - sender manages Sequence Number
  - receiver implements sliding window

Authentication and Reply Protection in ESP

- *Protection against replay attack:* same as in AH
  - sender manages Sequence Number
  - receiver implements sliding window
- *Integrity and authentication:* computed over ESP header (SPI and Sequence Number)

Confidentiality in ESP

- Covers payload
- Padding
  - match plaintext length required by encryption algorithm
  - additional confidentiality, by hiding the actual payload length
- *encryption algorithm:* implementations must support DES in CBC
- Other algorithms have been identified
  - 3DES
  - RC5
  - IDEA
  - 3IDEA
  - CAST
  - Blowfish

Summary of Services and Protocols

	AH	ESP authentication	ESP authentication + encryption
access control	•	•	•
connectionless integrity	•		•
data origin authentication	•		•
rejection or replayed packets	•	•	•
confidentiality		•	•
limited traffic-flow confidentiality		•	•

Security Policy Database (SPD)

- The decision is based on a ***security policy*** stored in a Security Policy Database (SPD)
- The SPD contains an ordered list of policy entries

$$selectors \Longrightarrow \{bypass, discard, IPSec\ processing\}$$

  - source address
  - destination address
  - protocol (IPv4 Protocol or IPv6 Next Header field)
  - source port
  - destination port
  - user id
  - ...

  Selectors may use single values, lists of values, or wild-card expressions.
- Each selector points to a ***security association***

Security Association

- A security association (SA) is *uni-directional*, and is identified by
  - destination address
  - a security parameter index (SPI)
  - a flag that says whether the association uses ESP or AH

- *SA* state (parameters stored in the SA database)
  - *sequence number counter:* 32-bit value used to generate sequence numbers in AH and ESP headers
  - *sequence counter overflow:* flag indicating whether an overflow of the sequence number counter should produce an auditable event
  - *anti-replay window*
  - *AH information:* authentication algorithm, keys, key lifetime, other AH-related parameters
  - *ESP information:* encryption and authentication algorithms, keys (encryption, and auth.), key lifetimes, initialization values, and other ESP-related parameters
  - *lifetime of the SA:* time interval or byte count after which an SA must be replaced with a new SA (and new SPI) or terminated
  - *IPSec protocol mode*
  - *path MTU*

Observations

- Key exchange and cipher negotiation is outside the scope of the basic architecture

Observations

- Key exchange and cipher negotiation is outside the scope of the basic architecture
- Policy definition is outside of the scope of IPSec

Observations

- Key exchange and cipher negotiation is outside the scope of the basic architecture
- Policy definition is outside of the scope of IPSec
- Policy implementation is essentially a network management issue

Observations

- Key exchange and cipher negotiation is outside the scope of the basic architecture
- Policy definition is outside of the scope of IPSec
- Policy implementation is essentially a network management issue
- Tunneling modes, especially in ESP with authentication allows you to set up virtual private networks

Observations

- Key exchange and cipher negotiation is outside the scope of the basic architecture
- Policy definition is outside of the scope of IPSec
- Policy implementation is essentially a network management issue
- Tunneling modes, especially in ESP with authentication allows you to set up virtual private networks
- Transport mode provides a security service comparable to transport-level security

Management of Security Associations

The general IPSec architecture specifies two classes of SA and key management:

- *Manual:* ad-hoc, network administrators directly configure the security databases and the policy databases.
- *Automated:* IKE: using Oakley, an authenticated key-exchange protocol

Management of Security Associations

The general IPSec architecture specifies two classes of SA and key management:

- *Manual:* ad-hoc, network administrators directly configure the security databases and the policy databases.
- *Automated:* IKE: using Oakley, an authenticated key-exchange protocol

Observations

- IPSec protocols (AH and ESP) are essentially *independent* of the way SAs are set up and maintained
- However, *the ultimate level of security of an IPSec association is almost completely dependent on the process by which the SA is setup and managed*
- Also, some specific features (e.g., the anti-replay counter feature) require an automated management procedure

Manual vs. Automated Management

Obvious advantages and disadvantages of manual management

- \+ ad hoc: easier to implement, at least initially
- – not scalable
- – unable to support some IPSec features (e.g., anti-replay in both AH and ESP)

Manual vs. Automated Management

Obvious advantages and disadvantages of manual management

- \+ ad hoc: easier to implement, at least initially
- – not scalable
- – unable to support some IPSec features (e.g., anti-replay in both AH and ESP)

and of automated management:

- \+ scalable to large networks, and across administrative boundaries
- – requires PKI for complete authentication

Automated SA and Key Management

The default automated key management protocol for IPSec is called *ISAKMP/Oakley.*

ISAKMP Internet Security Association and Key Management Protocol provides a framework for authentication and key exchange. ISAKMP is designed to be key exchange independent; that is, it is designed to support many different key exchange protocols

Oakley is a specific key-exchange protocol defined as the default key-exchange protocol in ISAKMP

Phase 1 is where the two ISAKMP peers establish a secure, authenticated channel with which to communicate. This is called the ISAKMP Security Association (not to be confused with the SAs that the protocol is trying to establish).

Phase 2 is where Security Associations are negotiated on behalf of services such as IPsec or any other service which needs key material and/or parameter negotiation

The ISAKMP SA is the shared policy and key(s) used by the negotiating peers in this protocol (phase 1) to protect their communication.

Overview of Oakley

Oakley is a refinement of Diffie-Hellman. Diffie-Hellman has the following problems that Oakley solves:

- No authentication
- Vulnerable to man-in-the-middle attack
- Computationally intensive, therefore vulnerable to DOS attack

Oakley uses the following features to counter these weaknesses:

- *Cookies* to counter clogging attacks
- *Groups* global parameters of the Diffie-Hellman key exchange (p and g). Oakley allows parties to negotiate groups
- *nonces* to protect against replay attacks
- *authenticated Diffie-Hellman* to protect against man-in-the-middle attack. Authentication is based on either:
  - digital signature (with public/private keys)
  - public-key encryption
  - symmetric-key encryption

ISAKMP Header

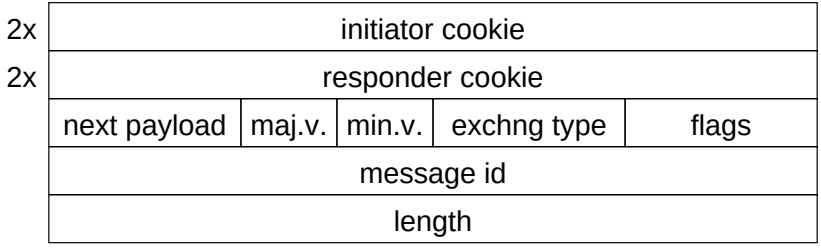
ISAKMP defines procedures and packet format (i.e., a protocol) to establish, negotiate, modify, and delete security associations.

In essence, ISAKMP defines a header.

ISAKMP Header

ISAKMP defines procedures and packet format (i.e., a protocol) to establish, negotiate, modify, and delete security associations.

In essence, ISAKMP defines a header.



This header introduces a *payload*

ISAKMP Payloads

Proposal payload used for SA negotiation

- SA protocol (AH or ESP)
- sender's SPI
- a list of *transform payloads*

Transform payload identifies a cryptographic function for a specific protocol function (e.g., 3DES for ESP), and its parameters.

Key exchange payload Oakley, or Diffie-Hellman, or RSA-based key exchange, etc.

Identification payload identifies the two ISAKMP peers

ISAKMP Payloads (3)

Certificate payload passes a public-key certificate

Hash payload verification (authentication) hash computed over some state information of this running ISAKMP

Nonce payload random data used during the exchange

Notification payload either error or status information associated with the current SA or with the SA negotiation

Delete payload identifies one or more SAs that the sender has deleted from its database

References:

- RFC 2408 Internet Security Association and Key Management Protocol (ISAKMP)
- RFC 2409 Internet Key Exchange (IKE)
- RFC 2407 Domain of Interpretation (DOI)

How Are Security Associations Established?

A security association can be established using an *ad-hoc* method—e.g., two system administrators can do that with pencil and paper over the phone—or through an authenticated key exchange protocol.

How Are Security Associations Established?

A security association can be established using an *ad-hoc* method—e.g., two system administrators can do that with pencil and paper over the phone—or through an authenticated key exchange protocol.

IPSec defines an authenticated key exchange protocol. More precisely: IKE/ISAKMP is a *framework* that allows several protocol variants. The general framework has two *phases*:

- *Phase 1* is an authenticated key exchange based on long-term keys. This phase establishes a session key
- *Phase 2* uses the session key established in phase 1 to define one or more security association

As we said, IKE is a framework that specifies various protocols. Two basic *modes* are defined:

- *aggressive mode* is a message-efficient scheme:
  1. $A : A, g^a \mod p, \text{crypto proposal} \longrightarrow B$
  2. $A \longleftarrow g^b \mod p, \text{crypto selection, proof I'm Bob} : B$
  3. $A : \text{proof I'm Alice} \longrightarrow B$
- *main mode*
  1. $A : \text{crypto proposal} \longrightarrow B$
  2. $A \longleftarrow \text{crypto selection} : B$
  3. $A : A, g^a \mod p \longrightarrow B$
  4. $A \longleftarrow g^b \mod p : B$
  5. $A : E_{k=g^{ab} \mod p}(\text{Alice, proof I'm Alice}) \longrightarrow B$
  6. $A \longleftarrow E_{k=g^{ab} \mod p}(\text{Bob, proof I'm Bob}) : B$