Part 2: Protocols supporting security mechanisms

- The theory of hacking is for the part of the lectures finished, continued in the exercises.
- We cannot treat in the short course all details, but basically - there is no theory of hacking, simply use bugs and human errors.
- The second part is about protocols supporting security. They use cryptographic protocols (key exchange etc.) as building components. Cryptographic protocols do not have all details of real life protocols, like packet formats, they are on idea level.
- We will look (most probably) at IPSec, Kerberos, SET, PGP.
- There are other relevant protocols, but we have to skip some.
- First we should bring back to mind some basic cryptographic concepts.

IPSec: Overview

- **Material:**
- The material for IPSec is from the book:
- N. Doraswamy, D. Harkins: IPSec, Prentice Hall, 1999.
- Kerberos and PGP are from the book
- Internet Security, a Professional guide, second edition
- and Secure Electronic Transactions from the book
- G. N. Drew: Using SET, Prentice Hall, 1999.
- The lecture notes suffice for the exam.
- **IPSec (for IPv4, IPv6 security features are similar)**
- Protects data in the Internet by adding authentication of data source (no IP address spoofing), preventing replays of old data, provides integrity of data (no modifications), and in some modes provides confidentiality of data.

IPSec: Cryptography basics

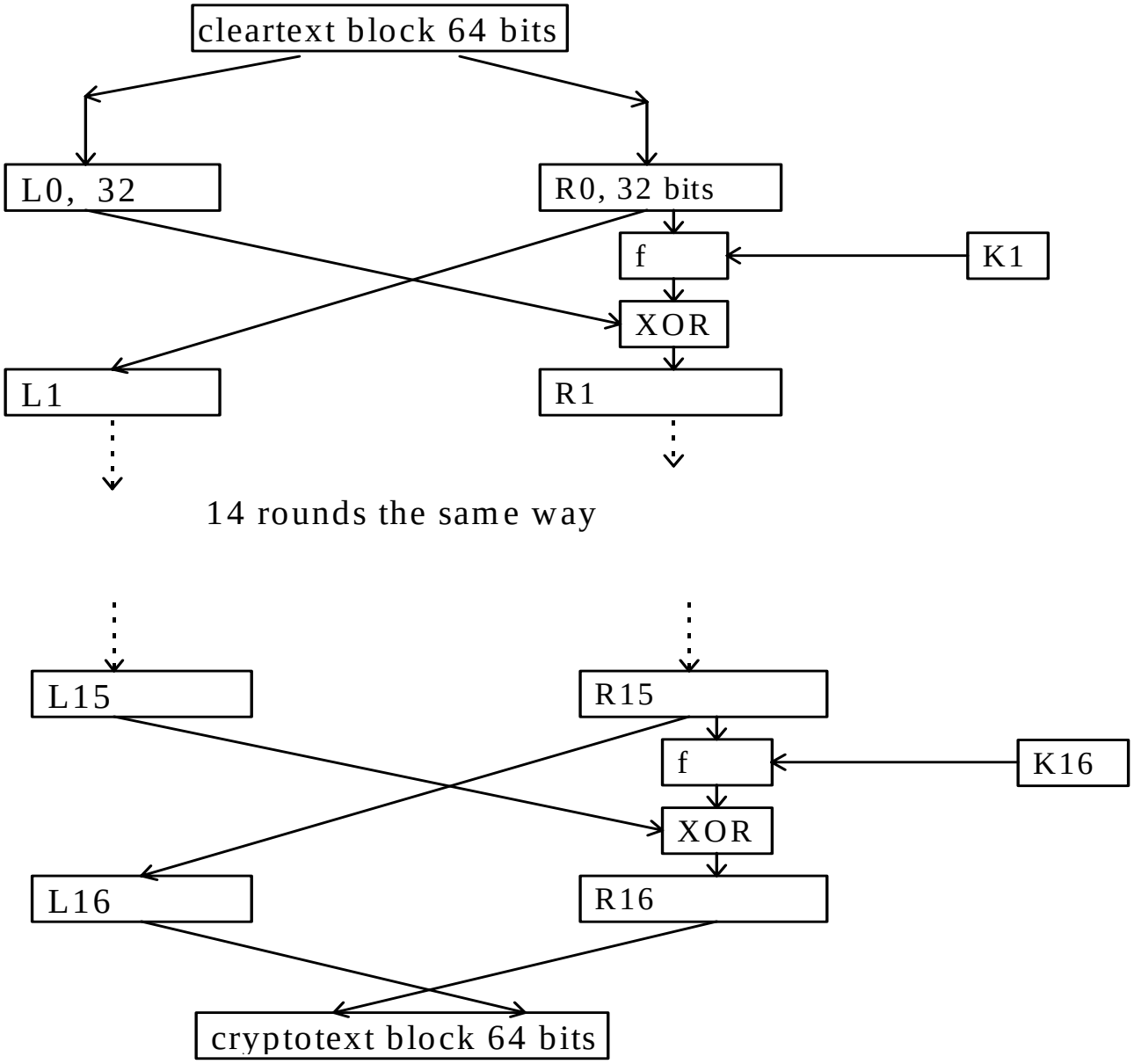
- IPSec is one of the main method which try to bring security to the Internet. IPSec is based on modern cryptographic methods. While this course is not explaining cryptography, some basic concepts should be refreshened.
- **Symmetric and asymmetric cryptoalgorithms**
- a symmetric cryptoalgorithm uses the same key in both sending and receiving side, asymmetric cryptoalgorithms, introduced by Diffie and Hellman 1976, use different keys to encrypt and to decrypt.
- **Stream and block ciphers**
- A stream cipher encrypts data working on each bit or byte separately, a block cipher encrypts a block (like 64 bits, 128 bits etc.) of data in one time.
- A stream cipher is fast but block ciphers are considered more secure. IPSec uses only block ciphers.

IPSec: Cryptography basics

- **DES Data Encryptation Standard**
- IPSec has a mandatory support for DES
- DES has 56 bit keys (expressed as 64 bit strings because of redundancy)
- DES is not any more safe, it can be broken in 20 hours with a special Deep Crack DES cracker.
- DES can be broken with linear cryptoanalysis in about 2^{47}
- steps, but Deep Crack cracks by brute force trying 2^{56} keys.
- DES is a Feistel network, meaning a special structure splitting a block (64 bits in DES) to two halves and mixing them so, that individual operations can loose information but the whole structure is bijective, so that you can crypt data and use the same Feistel network to decrypt it.

IPSec: Cryptography basics

- DES



IPSec: Cryptography basics

- In DES the plaintext block is divided into left and right blocks (L0, R0). The algorithms has 16 rounds and on each round the left and right blocks are swapped in the following way:

  Li = R(i-1) Ri = L(i-1) XOR f(R(i-1), Ki)

- So new left block is the previous right block and the new right block is obtained by XORing the previous left block with the previous right block encrypted with some function f using a key Ki.
- An encryptation algorithm satisfying this formula is a Feistel network. It means that f need not be a bijection for this encryptation to work. On each round f is a different function made with permutations and substitutions.
- The triple DES (3DES) has effective key length at least twice that of DES and is considered strong.

IPSec: Cryptography basics

- Other good symmetric block ciphers are IDEA, CAST and Blowfish. IPSec implementations have optional support for these algorithms. A main motivation for creation of a new standard is the ability to use longer block lengths than in DES.
- The new Advanced Encryptation Standard (AES) is recently elected. It is Rijndael.
- Rijndael has a flexible key size and flexible block length. 10 rounds on each round the cryptation function is a a simple combination of substitution and permutation.
- Rijndael is not a Feistel network, therefore on each round the encryptation function is bijective.
- It is possible to create strong cryptoalgorithms which are impossible to break, unless
- parallel computing methods like quantum computers are developed, (quite possible) or
- P=NP will be proved (unlikely, but possible)

IPSec: Cryptography basics

- **Asymmetric cryptoalgorithms**
- **RSA** (Rivest-Shamir-Adleman)

  find two large primes p, q and n=pq

  Find a number e such that e and (p-1)(q-1) are relatively prime (no common divisors).

  Find some d such that ed=1 mod n (this is easy)

  Then if X is a plaintext block, we get the ciphertext block Y

- $Y = X^e \bmod n$ and $X = Y^d \bmod n$
- **El-Gamal**
- Uses the discrete logarithm problem, quite similar to Diffie-Hellman key exchange algorithm. Encrypted block 2*plaintext block in length. Free. Before RSA patent expired (year 2000), El-Gamal was the favorite choice for US.

IPSec: Cryptography basics

- **Diffie-Hellman key exchange**
- IPSec Internet Key Exchange (IKE) uses Diffie-Hellman.
- Alice and Bob want to create a symmetric key for communication. So, they want to create a common secret which only they share. Let the generator number g and some prime number p be known to all (not secret).
- Alice picks up a number a and Bob picks up a number b.
- Then they calculate numbers A and B as

$$A = g^a \bmod p \qquad B = g^b \bmod p$$

- Alice sends to Bob the number A and Bob send to Alice number B. These numbers do not need to be kept secret.
- Alice and Bob can both count a shared secret S as

$$S = A^b \bmod p = B^a \bmod p = g^{ab} \bmod p$$

IPSec: Cryptography basics

- As asymmetric cryptoalgorithms are slow, usually one can only encrypt small data units with them.
- A common usage is digital signature: a hash value is produced by some one-way function which compresses the data. Then the hash value is crypted with a secret key.
- A one-way function is a function which is easy to calculate but difficult to invert, so it is easy to count the hash but difficult to find data which hashes to a given hash value.
- IPSec uses some well-known hash functions: MD5 and SHA.
- MD5 (Message Digest number 5) has some problems, one has demonstrated that it is possible to find two data values hashing to the same hash value.
- IPSec uses a strengthened version of the hash values: HMAC-MD5 does not have the problem.

IPSec: Cryptography basics

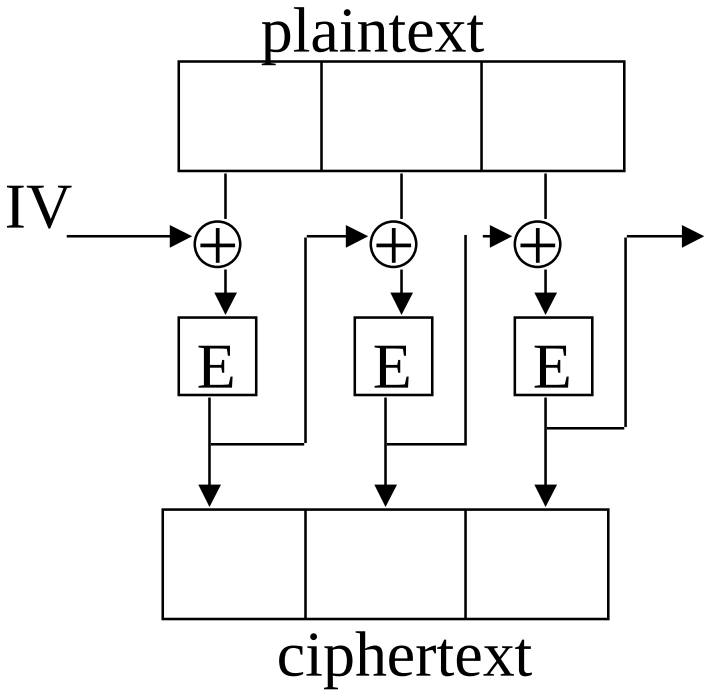
- **Digital signature algorithms used in IPSec:**
- **RSA,** suits well to digital signatures
- **DSA** (Digital Signature Algorithm), a similar algorithms to El-Gamal, uses SHA (Secure Hash Algorithm) for hashing.
- **Algorithms for message integrity in IPSec:**
- Digital signatures can be used to proof that the message has not changes. There are symmetric and asymmetric algorithms for this.
- **MAC** (Message Authentication Code) is a family of symmetric message integrity check algorithms. IPSec uses one special MAC: **HMAC**.
- It can be used with different hash functions, so there are HMAC-SHA and HMAC-MD5.

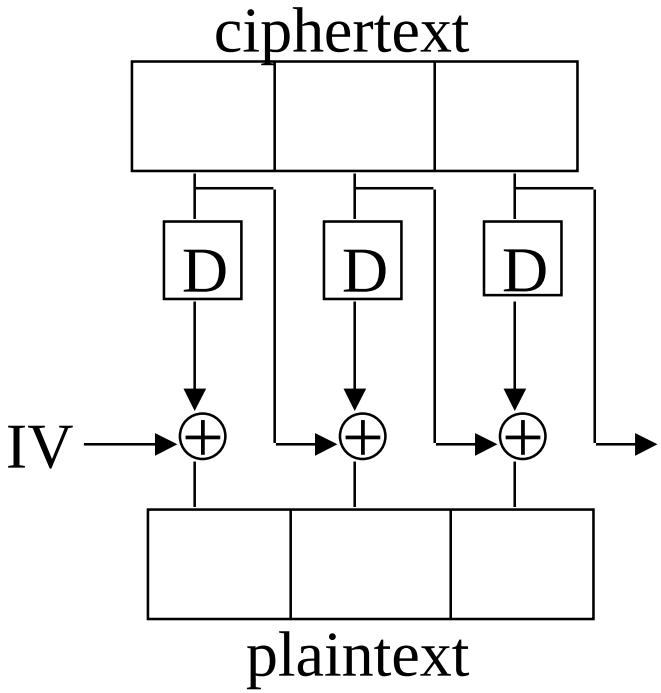
IPSec: Cryptography basics

- **Modes of symmetric cryptoalgorithms**
- Block ciphers can be used in several modes.
- Electronic Code Book (ECB), Cipher Block Chaining (CBC), Output Feedback (OFB), Counter mode, ...
- In ECB blocks are crypted individually, not suitable for communication, but good for storing data as then data can be decrypted without decrypting all previous blocks.
- In other modes previous plaintext or ciphertext blocks are used to encypt the next block.
- The feedback modes (CBC, OFB, Counter) differ mostly in error propagation. For links with high error ratio OFB or Counter mode are better than CBC.
- IPSec uses all block ciphers in the CBC-mode.

IPSec: Cryptography basics

- Cipher Block Chaining Mode (CBC)





IV = Initialization vector

E = Encryptation component

D = decryptation component

IPSec: Overview

- **On what layer to put security?**
- IPSec puts security mechanisms to the network layer. The question of if this is the correct layer is not easy.
- Many applications have security on application layer, like SSH, PGP, S-HTTP. The disadvantage is that applications need to be modified, but there are clear gains also
  - in some applications data may stay encrypted also in the computer like with PGP (with IPSec email will be plain text in the mailbox)
  - multiuser environment where several users use the network layer is easier to handle.
- Transport layer is a possible place for encryptation, like TLS (Transport Layer Security=Secure Socket Layer). The disadvantage is that TSL does not protect IP headers, but actually IP header protection is a problematic concept.

IPSec: Overview

- **On what layer to put security?**
- Link layer is the classical place to put encryptation. then each link connection is protected, usually with a stream cipher with symmetric keys.
- The Internet community considers link layer protection non-scalable. This is a misuse of words, the link layer encryptation technique is perfectly well scalable as protecting each link is a local mechanism.
- The problem is that the trust must be extended to all organizations operating link nodes, which usually is too much to trust.
- This mechanism is therefore used by networks owned by one organization, like a military network, and additionally end-to-end encryptation is applied on top of the link layer encryptation.

IPSec: Overview

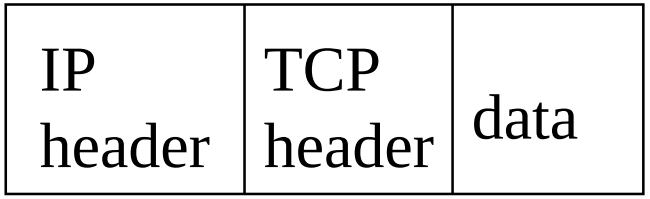
- **On what layer to put security?**
- Presentation layer or some common parts of an application layer are also possible places for security mechanisms.
- In the Internet OMG CORBA is a presentation layer concept and CORBA Services and Facilities are common application layer protocols. Some security mechanisms are put there, like CORBA Firewall, but encryptation with CORBA makes use of IPSec.
- One common argument for selecting the place where to put encryptation is that it is best to put encryptation on such a layer which is common to many protocols.
- In the OSI transport layer was a place where there were very few protocols (OSI-transport 0-4) and it was the main candidate for encryptation. In TCP/IP IP-layer or TCP/UDP-layer are the common protocols and the natural places for encryptation.

IPSec: Overview

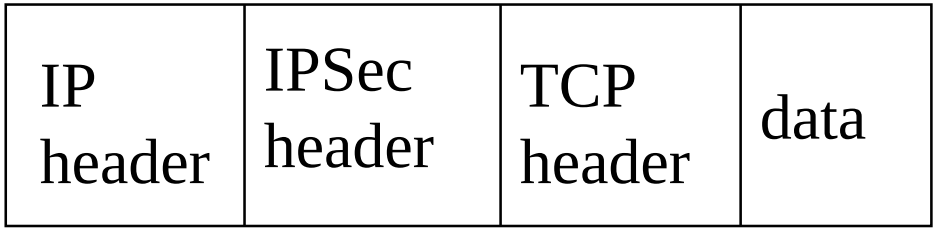
- **On what layer to put security?**
- The question is not the same for all aspects of security, authentication is easiest to do on the application layer.
- Protection against breaking into computers/network nodes is better done in the applications. The application bugs should be removed and access to applications should be restricted.
- Putting encryptation on the network layer is not without problems either:
- multicasting becomes more complicated,
- mobility support by the Mobile IP would require that an agent can read and redirect the first (IPv6 mobility) or all (mobile IP for IP v4) IP-packets. In the tunneling mode of IPSec the IP-packet header is encrypted and the addresses are not available, therefore the agent must be trusted.

IPSec: Overview

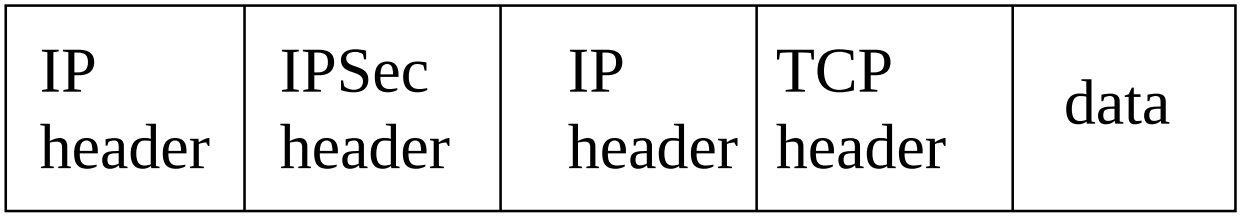
- Transport and Tunneling mode



original IP packet



transport mode



tunneling mode

The difference between the modes is what data they protect.
Transport mode protects transport (TCP, UDP) data.
Tunneling mode protects IP packets.

IPSec: Overview

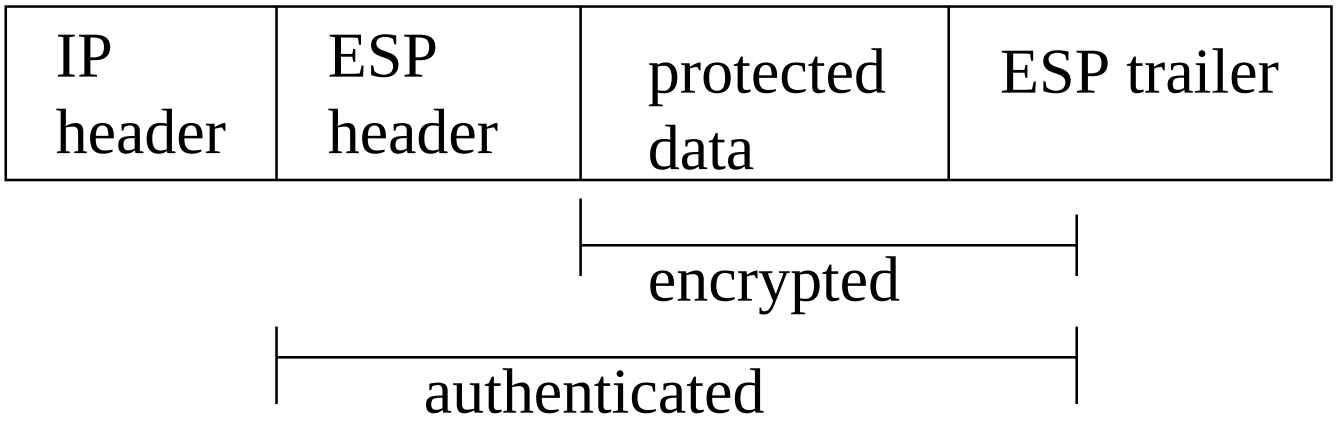
- **ESP and AH**
- IPSec has two protocols: Encapsulating Security Payload (ESP) and Authentication Header (AH).
- As ESP contains all features what AH contains and more, there is no clear reason why AH exists, but let us not comment on it.
- ESP is used both on transport and on tunneling mode.
- AH can operate on both modes, but it is used only on transport mode as tunneling mode for AH protects the same data as transport mode for AH.
- Security of ESP and AH depend on the cryptoalgorithms used, the default mode DES with CBC is not any more considered secure for sensitive data.
- Key management can be manual or based on IKE.

IPSec: Overview

- **Replay prevention by sequence number**
- IPSec protects against replay by sequence numbers and a sliding window protocol.
- IPSec packet header contains a monotonically increasing 32-bit sequence number. Wraps around after 2^{32} packets.
- The receiver window is any number bigger than 32, recommended window size is 64.
- Received packets must be either new (larger number than the received one) or not older than the receiver window size, else they are dropped.
- This enables receiving packets in changed order, provided that they are in the same window, and still detecting replayed packets.
- The window is advanced when the packet with the smallest number in the window is received and authenticated.

IPSec: Overview

- **ESP (Encapsulated Security Payload)**
- Provides proof-of-data origin for received packets, data integrity, antiplay protection and optionally data confidentiality.

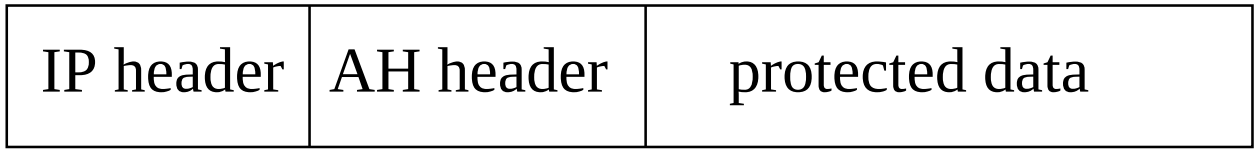


ESP header is not encrypted, part of ESP trailer is encrypted.

SPI (Security Parameter Index) and destination IP address must be in plaintext so that Security Association (SA) is identified. Sequence number and authentication field are in plaintext.

IPSec: Overview

- **AH (Authentication Header)**
- AH provides data integrity, data source authentication and protection against replays. No data confidentiality option.



AH header contains SPI. sequence number, authentication data field.

The authentication data field contains a digest of the MAC used to secure the data.

In both AH and ESP mandatory supported MACs are HMAC-MD5 and HMAC-SHA. The MAC fields are truncated to 96 bits. Truncation is for IPv6 compatibility and it is thought to be secure to truncate MACs (no proof).

IPSec: Overview

- Truncating MAC fields is necessary as different algorithms produce different length MAC values.
- EPS authentication (shown in the figure) does not cover the outer IP header.
- AH authentication covers the outer IP header of the packet. As there are IP header fields which change when the packet passes routers, these fields are set to zero before calculating the authentication data field.
- Other special features, like fragmentation and reassembly are also treated in AH documents.
- ESP has optional confidentiality, therefore ESP SA describes two algorithms: (cipher can be NULL)
- - cipher is for data confidentiality (DES-CBC is mandatory, Blowfish-CBC, CAST-CBC, 3DES-CBC are optional)
- - authenticator is the MAC algorithm.

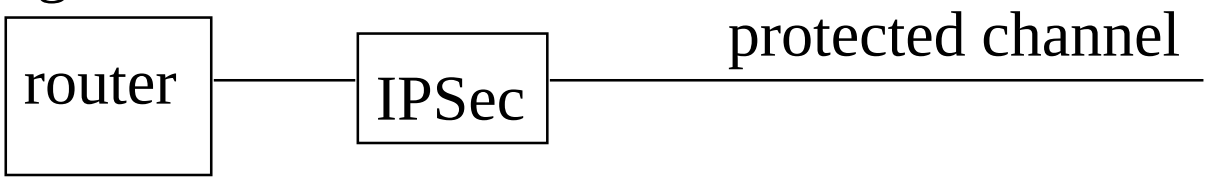
IPSec: Overview

- **OS and BITS implementation**
- IPSec is preferably implemented in the operating system level (OS) and merged with the IP level. This is called IPSec stack method. All IPSec options can be implemented and the implementation can be very efficient. IP fragmentation can be handled with the same code as IP uses.
- If it is not possible to mess up with the native IP implementation IPSec can be implemented as a separate layer between the Data Link layer and the Network (IP) layer, this is called Bump In The Stack (BITS) method.
- The BITS method is commonly used by Firewall providers who make software for equipment of many vendors and want to provide complete security solutions. Disadvantages are that fragmentation and some other network functions must be duplicated on the IPSec layer.

IPSec: Overview

- **BITW (Bump In The Wire)**
- This refers to a solution where a separate network equipment is inserted on the physical link and IPSec is provided there.

  A router solution enables securing data in the wild for organizations who trust their internal network.



  BITW method is not considered scalable, it is expected to be a transitory solution, the favorite choice is to mix IPSec with the router OS.

  Efficiency is a concern for IPSec. Putting IPSec in a router may slow down the router even though IPSec only deals with packets requiring security. Hard to say if it is a good idea to implement IPSec in a router at all.

IPSec: Overview

- **Security Association (SA) and other concepts**
- SA associates security services and a key with the traffic which is being protected. SA is unidirectional.
- SA is identified by SPI (Security Parameter Index), IPSec protocol value, and the destination IP address. Both pairs of communication have the SA, usually in the SADB (Security Association Database).
- SAs may be created manually or dynamically. Manually created SAs stay until, they are manually deleted. Dynamically created SAs have a lifetime which is negotiated by the key management protocol.
- SDP (Security Policy Database) defines what traffic is protected, how the traffic is protected, and with whom the protection is shared. SDP entry may specify: discard a packet, bypass it, or apply security mechanisms.

IPSec: Overview

- **IKE** (Internet Key Exchange) (we will continue on this later)
- Establishes shared security parameters and authenticated keys (that is SAs) between IPSec peers. However, IKE SA is not precisely IPSec SA as IKE can be used to negotiate SA for any protocol.
- DOI (Domain of Interpretation) defines how to use IKE, for IPSec DOI is defined by RFC 2407.
- IKE is a hybrid of Oakley and SKEME protocols.
- From Oakley IKE has taken 5 ways of exchanging secret, they are in phase one, creation of SA.
- Phase one ends with Diffie-Hellman key exchange, which IKE copied from SKEME. You can negotiate the parameters to Diffie-Hellman, but IKE always uses Diffie-Hellman for creation of a shared secret.

IPSec: Overview

- IKE supports two modes for phase one: Main Mode and Aggressive Mode.
- After the phase one comes authentication of the parties doing creation of SA. This is phase two.
- IKE has one mode for phase two: Quick Mode.
- There are five methods for authentication in IKE: preshared keys, digital signature using DSS, digital signature using RSA, encrypted nonce (random number) exchange using RSA, and the revised nonce method.
- In IKE packet formats, retransmission timers, message construction requirements are defined by ISAKMP standard (Internet Security Association and Key Management Protocol).

IPSec: Overview

- **Denial of Service protection**
- DoS attack can be directed to network elements implementing IPsec: since Diffie-Hellman key exchange is a heavy operation in processing time, a computer doing it can be overloaded by sending lots of bogus packets which first have to be authenticated (to see that they are bogus) before they can be discarded.
- In the Main Mode IKE protects against this by using a cookie method: a cookie is exchanged first, it is rather secure and fast to check, only if that matches, authentication is checked.
- Notice: DoS attack is not only overloading network elements, we can throw packets away. There is no way an unsecured network can be protected against throwing away packets by some misbehaving network element by cryptography, that is

  DoS cannot be removed in an unsecured physical environment.

IPSec: Overview

- **SA management**
- SA management takes care of creation and deletion of SAs. Management can be manual or using IKE.
- SA management updates SAs to SADB.
- In manual SA management users of IPSec agree on parameters using phone or email. Manual management is mostly useful in the debugging stage.
- Dynamically managed SAs are deleted when:
- lifetime has expired, keys are compromised,
- threshold number of bytes encrypted/decrypted by a key is exceeded, or
- the other end requests that SA is deleted
- when SA is deleted, SPI for it can be reused

IPSec: Overview

- IPSec summary