

Computer and Network Security

©Copyright 2000 R. E. Newman

Computer & Information Sciences & Engineering
University Of Florida
Gainesville, Florida 32611-6120
nemo@cise.ufl.edu

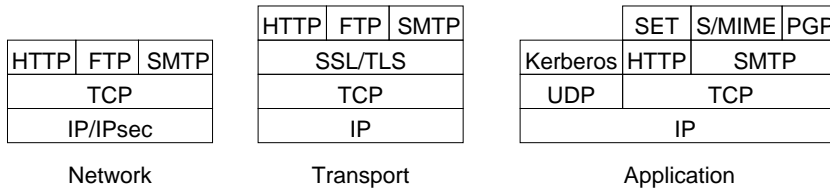
Network Security Protocols

Network Security Protocols

OSI, GSSAPI, IPSO, SSL

R. E. Newman
 nemo@cise.ufl.edu
 352-392-1488

Security Protocol Locations



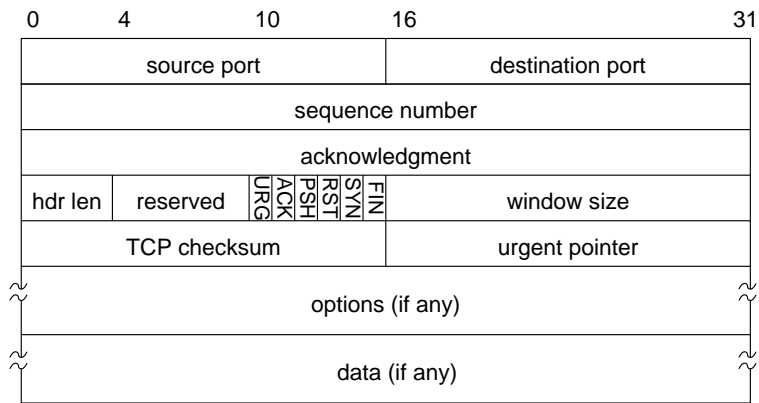
OSI Reference Architecture Layers

- 7 – Application
 - underlying OS security
 - interoperation
 - multiple applications
 - 6 – Presentation
 - compression, encryption, common coding
 - 5 – Session
 - managing multiple related streams
 - 4 – Transport
 - end-to-end
 - 3 – Network
 - host identification only
 - per-packet OH if datagram service
 - higher level entities are treated same
 - 2 – Link
 - 1 – Physical
- >
- only useful for immediate neighbors

OSI Security Services – Classes

1. Peer entity authentication service – no replay
Data origin authentication service – no modification/duplication prot.
2. Access control service
3. Connection confidentiality service
 - Connectionless confidentiality service
 - Selected field confidentiality service
 - Traffic flow confidentiality service
4. Connection integrity service with recovery
 - Connection integrity service without recovery
 - Selected field connection integrity service
 - Connectionless integrity service
 - Selected field connectionless integrity service
5. Non-repudiation with proof of origin
 - Non-repudiation with proof of delivery

TCP Packet Format

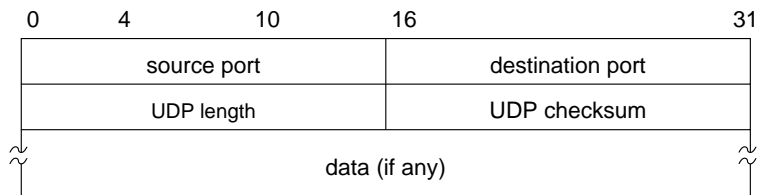


(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA

UDP Packet Format



(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA

IPSO – Security Association

Defining Parameters:

- Authentication algorithm and algorithm mode in AH (req)
- Key(s) used with authentication algorithm (req)
- Encryption algorithm, mode and transform in ESP (req)
- Key(s) for ESP (req)
- Flag and size of crypto synch or IV for ESP (req)
- Authentication algorithm and mode for ESP (rec)
- Authentication keys for ESP (rec)
- Key lifetime (rec)
- SA lifetime (rec)
- Source address(es) of SA (wildcard if >1 source) (rec)
- Sensitivity level of protected data (req. for MLS, rec o/w)
(see RFC 1108 – DoD Security Options – labels
and whose rules are to be used for protection)

IPSO – Key Management

Manual – sysadmin configures keys

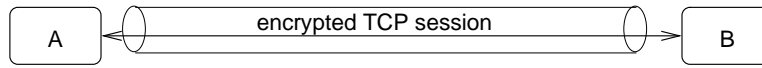
Automated – on-demand key creations for SAs, support of large systems

ISAKMP/Oakley – default automated key management protocol

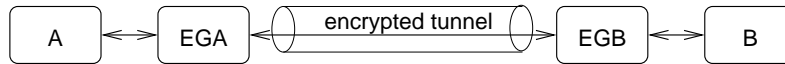
Oakley – Key determination protocol – based on Diffie–Hellman

ISAKMP – Internet Security Association and Key Management Protocol
framework for key management
provides specific formats, negotiation protocols

IPSO – Encapsulating Security Payload



Transport Mode



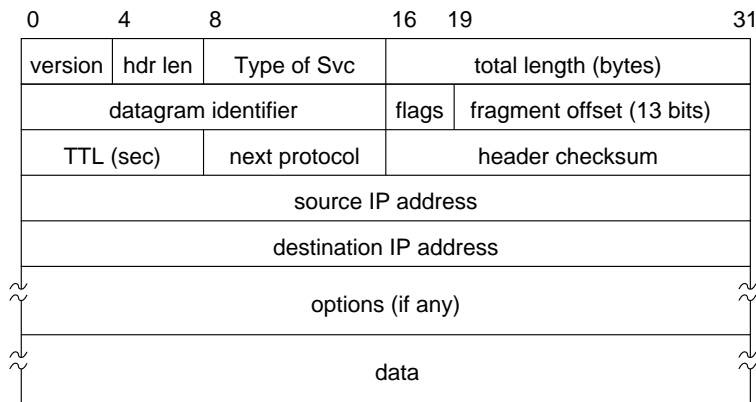
Tunnel Mode

(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA

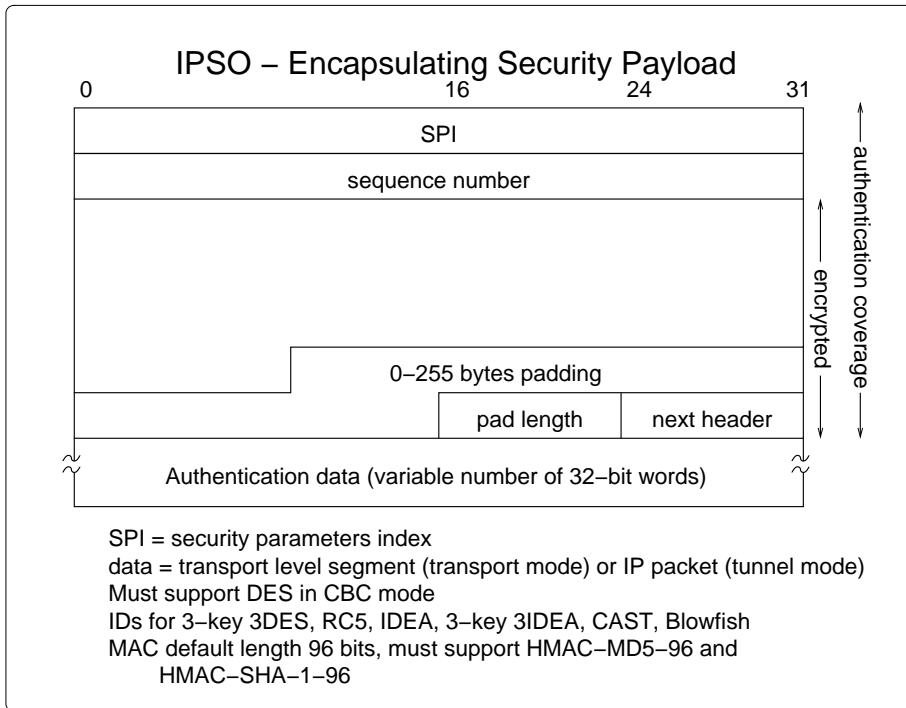
IP Packet Format



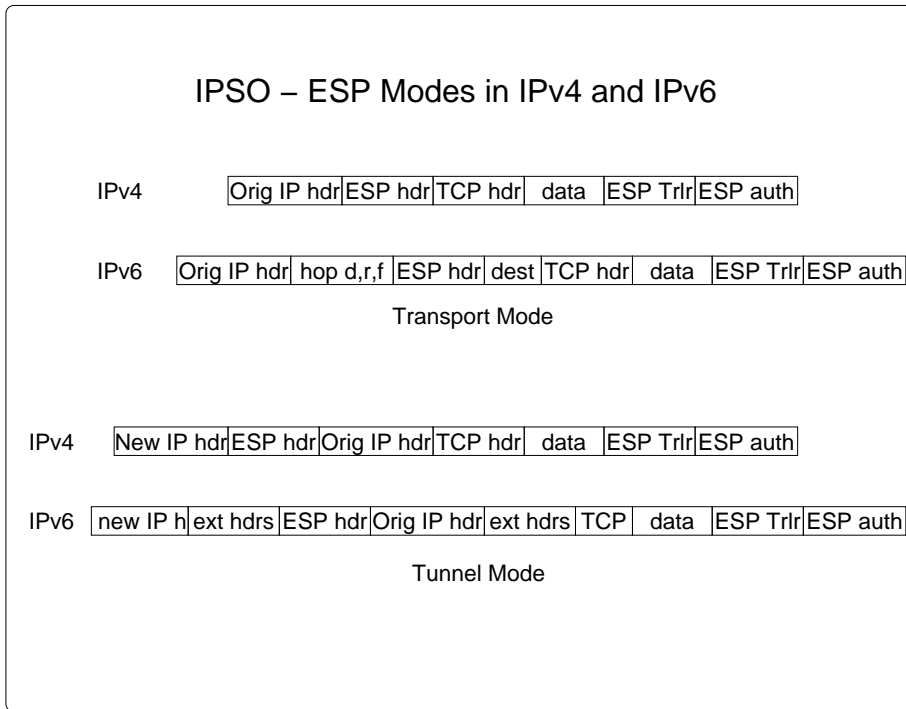
(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA



(c) copyright 1998 Richard E. Newman Department of Computer & Information Science & Engineering University of Florida, Gainesville, FL USA



(c) copyright 1998 Richard E. Newman Department of Computer & Information Science & Engineering University of Florida, Gainesville, FL USA

IPSO – ISAKMP

Establish, negotiate, modify, delete IPSO SAs

Defines payloads for key generation and authentication data independent of particular key exchange protocol, encryption algorithm, or authentication protocol

Five default exchanges –

- Base exchange (4) – key and authentication data sent together – gives Id
- ID protection (6) – protects IDs by establishing SA first, then key, then auth
- Authentication only (3) – establish AH SA
- Aggressive (3) – authentication and key exchange, reveals IDs
- Informational (1) – one-way transmission of info for SA management

IPSO – ISAKMP Payload Types

SA payload – begin establishment of SA – DOI – situation (reqts)

Proposal payload – SA negotiation – ESP/AH – SPI – # transforms

Transform payload – Transform # – Transform ID – parameters
defines specific transform, e.g., 3DES, HMAC–MD5–96

Key Exchange payload – KE data – depends on key exchange algorithm

Identification payload – ID data – usually IPv4 or IPv6 address

Certificate payload – transfers public key certificate (e.g., PGP, X.509,...)

Certificate Request payload – acceptable types, CAs

Hash payload – for integrity, authentication

Signature payload – digital signature data – integrity, nonrepudiation

Nonce payload – for liveness, replay prevention

Notification payload – error or status information

Delete payload – one or more SAs that are no longer valid

IPSO – Oakley Key Determination Protocol

Based on Diffie–Hellman

Uses cookies to defeat Denial of Service attacks

- start by sending cookies before DH exponentiation work
- can only force target to send ACKs on spoofed sessions
- cookies depend on source, dest IP/port and secret values

Supports groups (preset g , n for DH exchange)

Uses nonces to prevent replays

Enables DH public key value exchange

Authenticates DH key determination to prevent bucket–brigade attack

- digital signatures
- public key encryption
- symmetric key encryption

Supports various sequences for key exchange (e.g., aggressive, 3–msg)

Generic Security Service API (GSS–API)

RFC–1508: GSS–API RFC–1509: GSS–API bindings for C

- Provide interoperability for systems with different sets of security mechanisms
- Be independent of communication protocol, protocol association constructs

GSS_{Acquire|Release|Inquire}_cred() – credential management

GSS_{Init|Accept|Delete}_sec_context()
GSS_Process_context_token() – context management
GSS_Context_time()

GSS_{Sign|Verify|Seal|Unseal}() – per–message calls

GSS_Display_status() – support calls
GSS_Indicate_mechs()
GSS_{Compare|Display|Import|Release}_name()
GSS_Release_{buffer|oid_set}()

GSS–API caller accepts (uninterpreted) tokens provided by local GSS–API implementation, sends these to remote peer, who passes them to its impl.
Calls return a "more" or "done" status to indicate if further processing needed.

Secure Socket Layer Protocol Stack

SSL Handshake	SSL Change Cipher Spec	SSL Alert	HTTP
SSL Record protocol			
TCP			
IP			

Connection – OSI transport; peer-to-peer, transient relationship
Each connection associated with one session

Session – client-server association created by Handshake Protocol
Set of cryptographic security parameters to be used over many connections

SSL Connection State

Client and Server Random – random bytes chosen for each connection

Server write MAC secret – secret key used by server in MACs

Client write MAC secret – secret key used by client in MACs

Server write key – symmetric key used by server to encrypt data

Client write key – symmetric key used by client to encrypt data

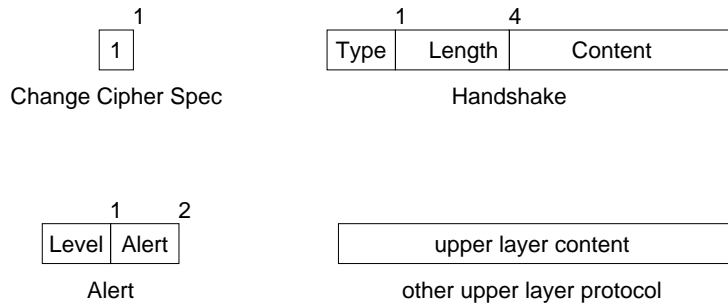
Initialization vectors – IV for each key used with CBC mode

Initialized by Handshake Protocol, then last ciphertext block
from each record used as IV for next record

Sequence numbers – each party has its own 64-bit sequence numbers

Change cipher spec messages (sent or received) reset these to 0
Sequence numbers may not wrap around

SSL Record Protocol Payload Formats



(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA

SSL Record Protocol

Provides –

- Confidentiality (SSL Handshake Protocol defines symmetric keys)
- Message Integrity (Handshake Protocol defines MAC secrets)

Steps –

- Fragment – into 16384-byte (max) chunks
- Compress – lossless, no expansion more than 1024 bytes (default null)
- Add MAC – $H(\text{MWS} \mid \text{pad-2} \mid H(\text{MWS} \mid \text{pad-1} \mid \text{seq \#} \mid \text{SSL-comp-type} \mid \text{SSL-comp-length} \mid \text{SSL-compressed-fragment}))$
- Encrypt – may not increase length by more than 1024 bytes
- Prepend header – Content type (8), Major version (8), Minor version (8), Compressed length (16 bits)
Content type is change_cipher_spec, alert, handshake, or application_data

(c) copyright 1998 Richard E. Newman

Department of Computer & Information Science & Engineering

University of Florida, Gainesville, FL USA

SSL Session State

- Session ID – for server to identify active or resumable state info
- Peer certificate – X.509.v3 certificate of peer (may be null)
- Compression method – for pre-encryption compression
- Cipher spec – bulk encryption algorithm, hash algorithm, other parameters (e.g., hash length)
- Master Secret – 48-byte secret shared between client and server
- Is-resumable Flag – indicates whether the session may be used to initiate new connections

SSL Handshake Protocol

