Secure Sockets Layer (SSL)

- Transport layer security service
  - originally developed by Netscape
  - version 3 designed with public input
- Subsequently became Internet standard RFC2246: Transport Layer Security (TLS)
- Uses TCP to provide a reliable end-to-end service
- may be provided in underlying protocol suite, or embedded in specific packages

SSL Protocol Stack

SSL Record Protocol Services

- Message integrity
  - using a MAC with shared secret key
  - similar to HMAC but with different padding
- Confidentiality
  - using symmetric encryption with a shared secret key defined by Handshake Protocol
  - Algorithms negotiated: AES, IDEA, etc.
  - Message is compressed before encryption

SSL Record Protocol Operation
**SSL Handshake Protocol**

- Allows server and client to:
  - authenticate each other
  - negotiate encryption and MAC algorithms
  - negotiate cryptographic keys to be used
- Comprised of a series of messages in phases
  1. Establish Security Capabilities
  2. Server Authentication and Key Exchange
  3. Client Authentication and Key Exchange
  4. Finish

**SSL Change Cipher Spec Protocol**

- One of three SSL specific protocols which use the SSL Record protocol
- A single message
- Causes pending state to become current, hence updating the cipher suite in use

**SSL Alert Protocol**

- Conveys SSL-related alerts to peer entity
- Severity: warning or fatal
- Specific alert
  - fatal: unexpected message, bad record MAC, decompression failure, handshake failure, illegal parameter
  - warning: close notify, no certificate, bad certificate, unsupported certificate, certificate revoked, certificate expired, certificate unknown
- Compressed and encrypted like all SSL data

**IP Security**

- Various application security mechanisms *e.g.* S/MIME, PGP, Kerberos, SSL/HTTPS
- Security concerns cross protocol layers
- Hence, we would like security implemented by the network for all applications
- Authentication and encryption security features included in next-generation IPv6
- They’re also usable in existing IPv4

**IPSec**

- General IP Security mechanisms
- Provides
  - authentication
  - confidentiality
  - key management
- Applicable to use over LANs, across public and private WANs, and for the Internet
### IPSec Uses

![IPSec Uses Diagram]

- In a firewall/router, provides strong security to all traffic crossing the perimeter
- In a firewall/router, is resistant to bypass
- Is below transport layer, hence transparent to applications
- Can be transparent to end users
- Can provide security for individual users
- Secures routing architecture

### Benefits of IPSec

- In a firewall/router, provides strong security to all traffic crossing the perimeter
- In a firewall/router, is resistant to bypass
- Is below transport layer, hence transparent to applications
- Can be transparent to end users
- Can provide security for individual users
- Secures routing architecture

### IP Security Architecture

- Mandatory in IPv6, optional in IPv4
- There are two security header extensions:
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)
- Key Exchange function
- VPNs want both authentication and encryption, hence usually use ESP
- The specification is complex; described in numerous RFC’s: 2401/2402/2406/2408

### Security Associations

- A one-way relationship between sender and receiver that affords security for traffic flow
- Defined by 3 parameters:
  - Security Parameters Index (SPI)
  - IP Destination Address
  - Security Protocol Identifier
- Has a number of other parameters
  - sequence number, authentication header and encryption header information, lifetime, etc.
- An implementation requires a database of Security Associations

### Authentication Header (AH)

- Provides support for data integrity and authentication of IP packets
- end system/router can authenticate user/app
- prevents address spoofing attacks by tracking sequence numbers
- Based on use of a MAC: HMAC-MD5-96 or HMAC-SHA-1-96
- Parties must share a secret key

### Authentication Header

![Authentication Header Diagram]
Encapsulating Security Payload (ESP)

Key Management
- Handles key generation and distribution
- You typically need 2 pairs of keys; one key per direction for AH and ESP
- Manual key management: system administrator manually configures every system
- Automated key management: An automated system for on demand creation of keys for SA’s in large systems

S/MIME (Secure/Multipurpose Internet Mail Extensions)
- A security enhancement to MIME email
  - original Internet RFC822 email was text only
  - MIME provided support for varying content types and multi-part messages
  - added encoding of binary data to textual form
  - S/MIME added security enhancements
- There is S/MIME support in many mail agents: MS Outlook, Mozilla, Mac Mail, etc.

S/MIME Functions
- Enveloped data: encrypted content and associated keys
- Signed data: encoded message + signed digest
- Clear-signed data: cleartext message + encoded signed digest
- Signed and enveloped data: nesting of signed and encrypted entities

S/MIME Process

S/MIME Cryptographic Algorithms
- Digital signatures: DSS and RSA
- Hash functions: SHA-1 and MD5
- Session key encryption: El Gamal and RSA
- Message encryption: AES, 3DES, etc
- MAC: HMAC with SHA-1
- Must map binary values to printable ASCII: use radix-64 or base64 mapping
S/MIME Public Key Certificates
• S/MIME has effective encryption and signature services
• But also needs to manage public keys
• S/MIME uses X.509 v3 certificates
• Each client has a list of trusted CA’s certs
• And also its own public/private key pairs and certificates
• Certificates must be signed by trusted CA’s

Identification and Authentication
• Identification?
  WHO ARE YOU?
• Authentication?
  PROVE IT!
• Authorization
  WHAT CAN YOU DO?

On-Line Identity
On the Internet, nobody knows you’re a dog.

Copyright © 1993, The New Yorker

Internet Authentication Applications
• Application-level authentication and digital signatures
• Implementations:
  • Kerberos symmetric key authentication service
  • X.509 public-key directory authentication
  • Public-key infrastructure (PKI)
  • Federated identity management

Kerberos
• Trusted key server system from MIT
• Provides centralised secret-key third-party authentication in a distributed network
  • Allows users access to services distributed through network…
  • …without needing to trust all workstations
  • Instead all trust a central authentication server
• Two versions in use: 4 and 5

Kerberos Overview
• A basic third-party authentication scheme
• Two servers (possibly one one machine)
• Authentication Server (AS)
  • users initially negotiate with AS to identify self
  • AS provides a non-corruptible authentication credential (ticket granting ticket TGT)
• Ticket Granting Server (TGS)
  • users subsequently request access to other services from TGS on basis of users TGT
Kerberos Overview

Kerberos Realms
- A Kerberos environment consists of:
  - a Kerberos server
  - a number of clients, all registered with server
  - application servers, sharing keys with server
- This is called a realm
  - typically a single administrative domain
  - For multiple realms, their Kerberos servers must share keys and trust

Kerberos Realms

Kerberos Overview

Kerberos Realms

Kerberos Version 5
- Kerberos v4 is most widely used version
- Also have v5, developed in mid 1990’s
  - specified as Internet standard RFC 1510
- Provides improvements over v4
  - addresses environmental shortcomings
    - encryption algorithm, network protocol, byte order, ticket lifetime, authentication forwarding, inter-realm authentication
  - and technical deficiencies
    - double encryption, non-std mode of use, session keys, password attacks

Kerberos Performance
- Works with larger client-server installations
- Kerberos performance impact is very little if system is properly configured, since tickets are reusable
- Kerberos security is best assured if the server is a separate, isolated machine
- Motivation for multiple realms is administrative, not performance

Certificate Authorities
- A digital certificate consists of:
  - a public key plus ID of the key owner
  - signed by a third party trusted by community
  - often government/bank certificate authority (CA)
  - Goal: bind an identity to a public key
- Users obtain certificates from CA
  - User creates keys and unsigned certificate, gives to CA
  - CA signs certificate, returns to user
- Other users can verify certificate by checking signature on certificate using CA’s public key
**X.509 Authentication Service**

- Universally accepted standard for formatting public-key certificates
- Widely used in network security applications, including IPSec, SSL, SET, and S/MIME
- Part of CCITT X.500 directory service standards
- Uses public-key cryptography and digital signatures
  - Algorithms not standardised, but RSA recommended

**X.509 Certificates**

- Functions:
  - Registration
  - Initialization
  - Certification
  - Key pair recovery
  - Key pair update
  - Revocation request
  - Cross certification
  - Protocols: CMP, CMC

**Public Key Infrastructure**

**PKI Management**

**Federated Identity Management**

- Definition: use of a common identity management scheme:
  - Across multiple enterprises and numerous applications
  - Supporting many thousands, even millions of users
- Principal elements are:
  - Authentication, authorization, accounting, provisioning, workflow automation, delegated administration, password synchronization, self-service password reset, federation
- Kerberos contains many of these elements

**Identity Management**
Federated Identity Management

Standards Used

- **Extensible Markup Language (XML)**
  - characterizes text elements in a document on appearance, function, meaning, or context
- **Simple Object Access Protocol (SOAP)**
  - for invoking code using XML over HTTP
- **WS-Security**
  - set of SOAP extensions for implementing message integrity and confidentiality in Web services
- **Security Assertion Markup Language (SAML)**
  - XML-based language for the exchange of security information between online business partners

Questions