Network Security Architectures Part 1 Fundamentals

Summer School on Software Security Theory to Practice

Carl A. Gunter
University of Pennsylvania
Summer 2004

Public Key Infrastructure

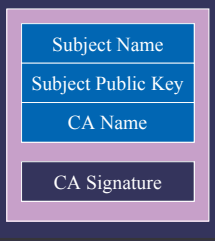
- Mutual authentication of participants in a transaction requires a system of identities
- *Principals* are identified by public keys
- These keys can be used for authentication, but only if "spoofing" is prevented
- A Public Key Infrastructure (PKI) provides a basis for establishing trust

PKI Systems

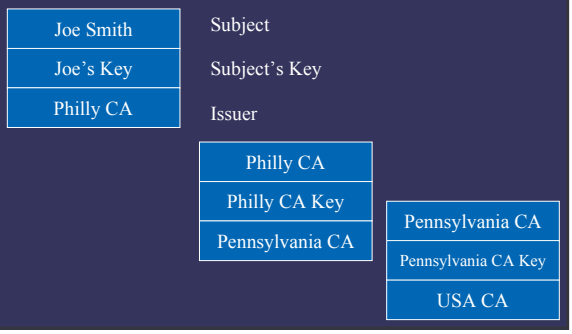
- Three Philosophies
  - Hierarchy
    - ITU X.509 (DAP, PKIX)
    - DNS
  - Web of Trust
    - PGP
  - Ad hoc
    - SSH
    - Most research studies

X.509 Certificates

X.509 certificates bind a subject to a public key.
This binding is signed by a *Certificate Authority (CA).*



Chaining



Certificate Management

- Distribution: How to find a certificate
  - Certificate accompanying signature or as part of a protocol
  - Directory service
    - DAP
    - LDAP
    - DNS
  - Email
  - Cut and paste from web pages
- Revocation: Terminate certificates before their expiration time.
  - How does the *relying party* know that the certificate has been revoked?
  - Many CRL distribution strategies proposed
  - Mitre report for NIST suggests certificate revocation will be the largest maintenance cost for PKIs

Semantics of CRL's

- Three certificates.

Revoke
1. Q says P is the public key of Alice.
2. R says P is the public key of Alice.
3. Q says R is the public key of Bob.

- Three kinds of revocation.
  1. P is not the public key of Alice. (3 not 2.)
  2. Q no longer vouches for whether P is the public key of Alice. (2 and 3.)
  3. The key of Q has been compromised. (2 not 3.)

Adoption of PKI

- Problems
  - Revocation
  - User ability to deal with keys
  - Registration (challenge for all authentication techniques)
  - Weak business model
- Areas of Progress
  - SSL
  - Authenticode
  - SSH
  - Smart cards for government employees
  - Web services

Challenges for Network Security

- Sharing
- Complexity
- Scale
- Unknown perimeter
- Anonymity
- Unknown paths

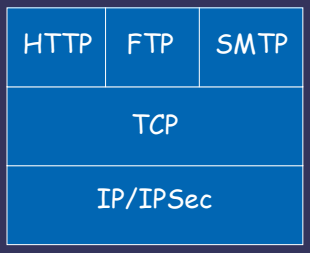
Internet Layers

1. Physical
2. Link
3. Network
4. Transport
5. Application

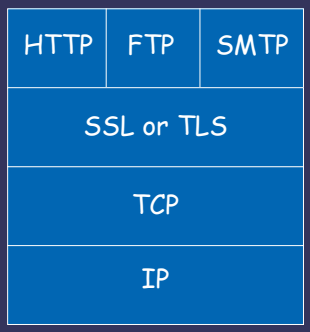
Security at Layers

- Physical
  - Locked doors
  - Spread spectrum
  - Tempest
- Link
  - WEP
  - GSM
- Network
  - Firewalls
  - IPSec
- Transport
  - SSL and TLS
- Application
  - S/MIME
  - XMLDSIG and WS security
  - Access control systems for web pages, databases, and file systems

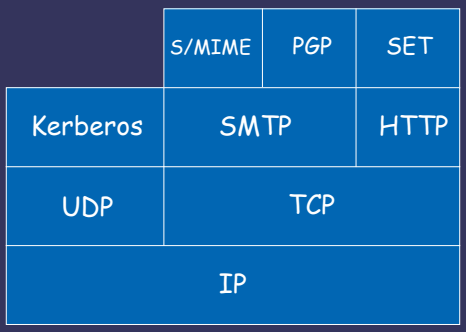
Network Layer Security



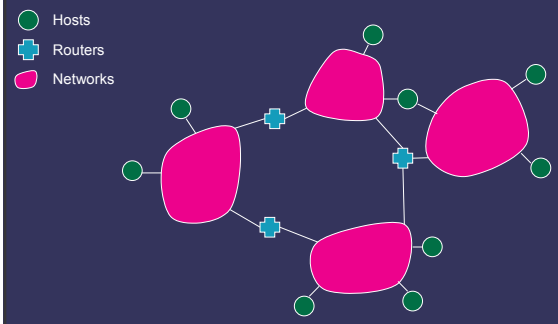
Transport Layer Security



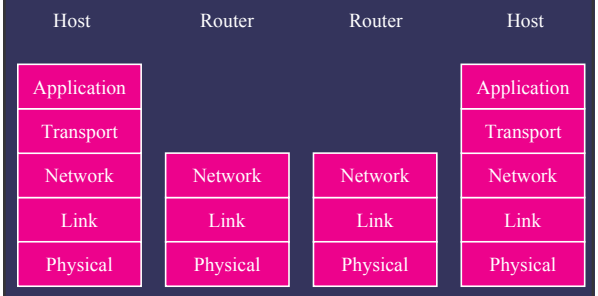
Application Layer Security



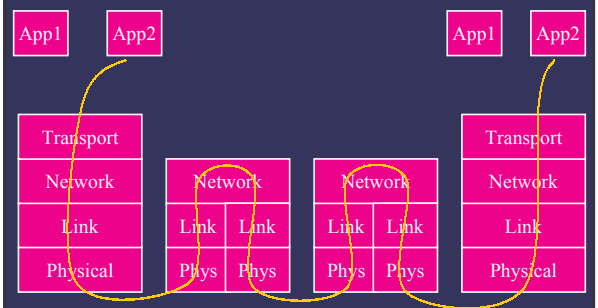
Division of Labor in the Internet



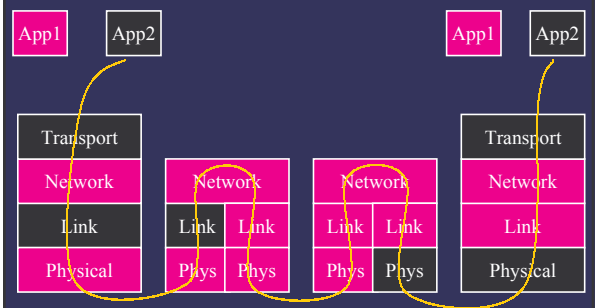
TCP/IP Protocol Stack



Communication Processing Flow



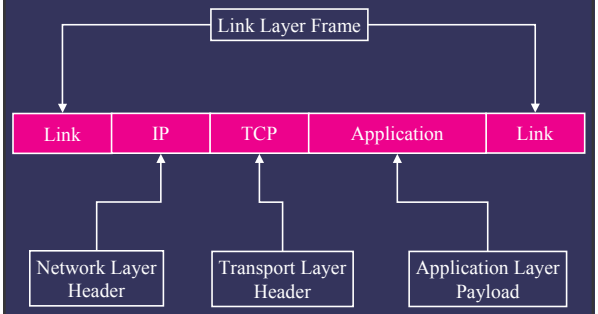
Typical Patchwork



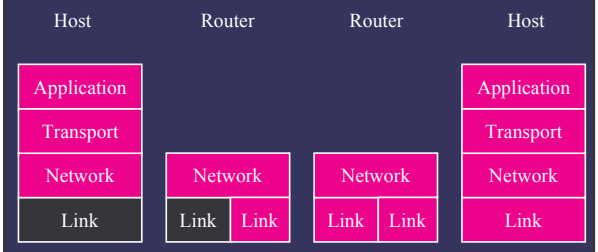
Physical Layer Protection Issues

- Hide signal
  - Spread spectrum
- Emission security
  - Radio emissions (Tempest)
  - Power emissions

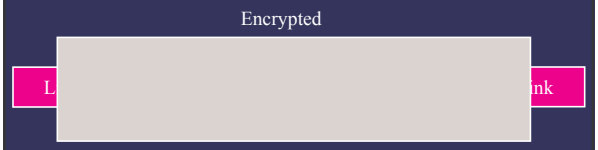
Encapsulation



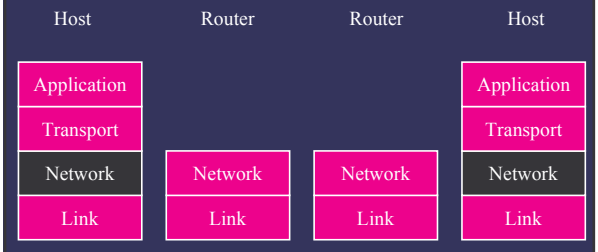
One Hop Link Layer Encryption



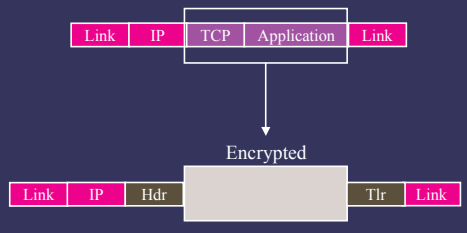
Link Layer Encryption



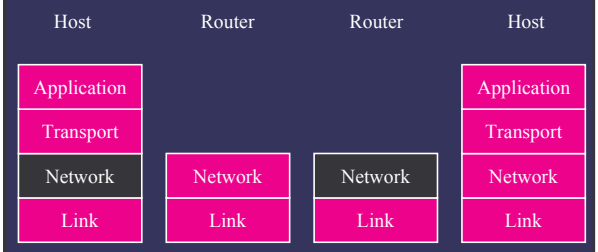
End-to-End Network Security



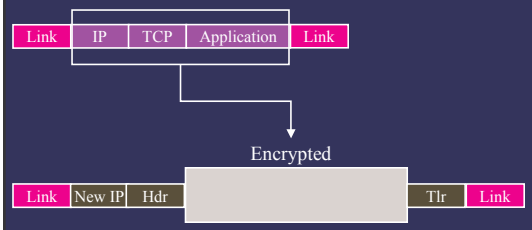
Network Layer Transport Mode



VPN Gateway



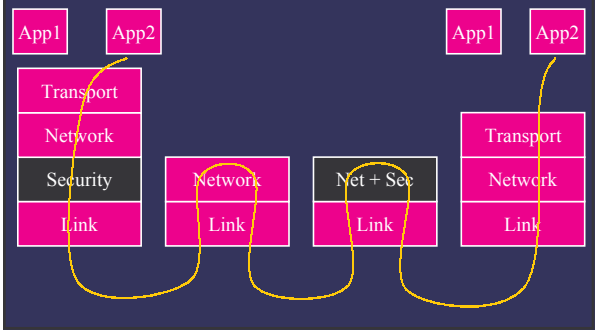
Network Layer Tunnel Mode



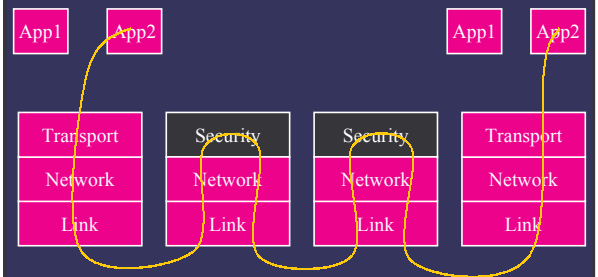
Layer 3 Implementation Options

- Location
  - Host
  - Network
- Style
  - Integrated
  - Modular (for tunnel mode)

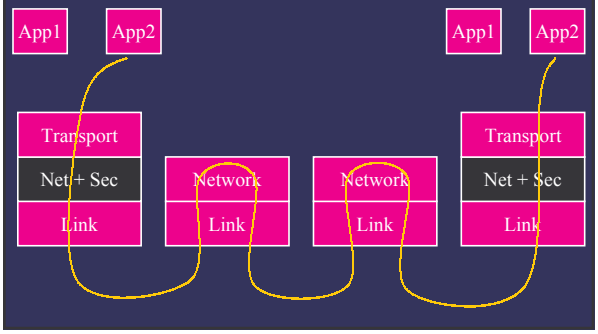
Modular Implementation: Bump In The Stack (BITS)



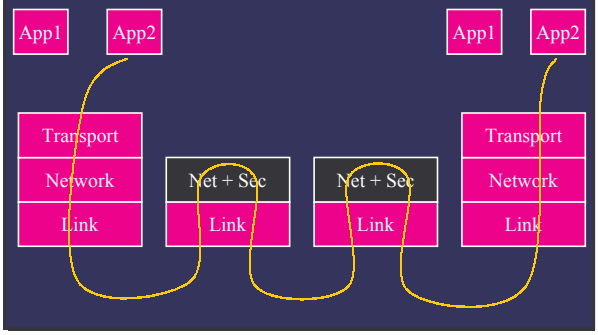
Modular Implementation: Bump In The Wire (BITW)



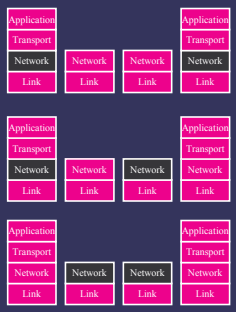
Implementation Options: Integrated on Host



Implementation Options: Integrated on Router



Network Security Location Options

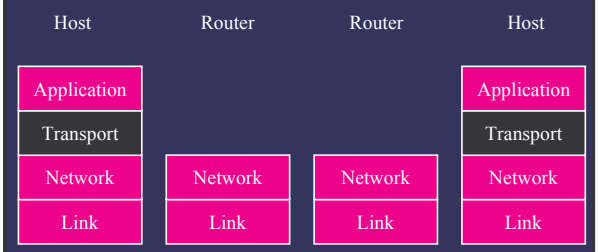


End-to-End Transport

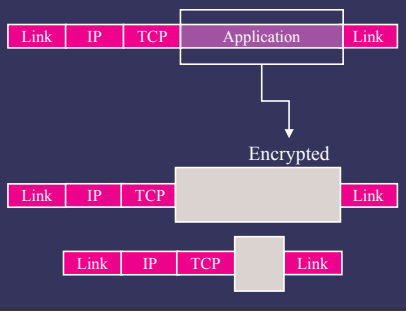
Voluntary Tunnel

Involuntary Tunnel

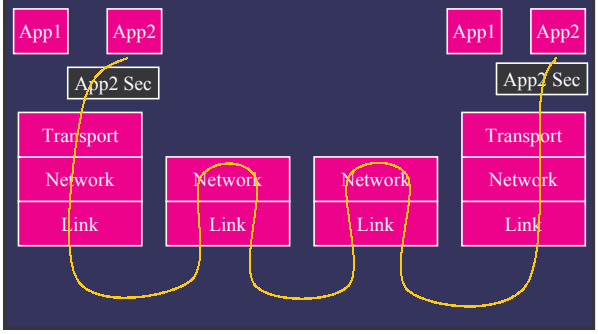
Transport Layer Security



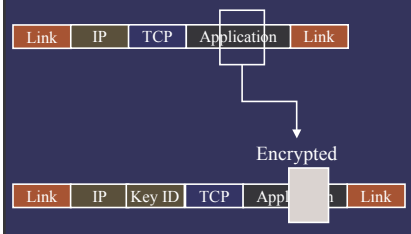
Transport Layer Encryption



Message Processing Sequence



Application Layer Security



Link Layer Security

- Advantages:
  - Transparent to applications
  - Hardware solution possible
  - Can address especially vulnerable links (viz. wireless)
- Disadvantages:
  - Hop-by-hop protection causes multiple applications of crypto operations
  - May not provide end to end security

Network Layer Security

- Advantages
  - Transparent to applications
  - Amenable to hardware
  - Flexible
- Disadvantages
  - Makes routing more complex
  - Flexibility introduces policy management and compatibility challenges

Transport Layer Security

- Advantages
  - Transparent to applications and may be packaged with applications
  - Exposing TCP enables compression and QoS classification
- Disadvantages
  - Probably implemented in software
  - Exposing TCP risks DoS

Application Layer Security

- Advantages
  - Customized to application
  - Requires no special protocol stack (transparent to networking)
- Disadvantages:
  - Hard to share between applications (viz. standardization challenge)

Protocols to Software

- There are important differences between theoretical descriptions, standards and software
  - Evolution (versions, extensibility)
  - Interoperability (options, negotiation)
  - Error modes
- Two brief case studies
  - Transport Layer Security (TLS)
  - Network layer security (Ipsec)

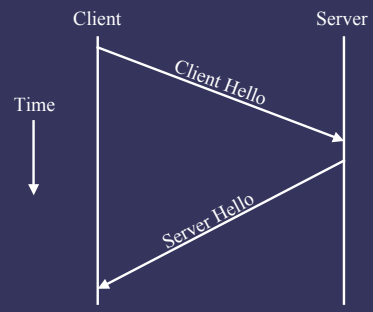
Secure Socket Layer (SSL)

- Session protocol with:
  - Server authentication
  - Client authentication optional
  - Integrity checksum
  - Confidentiality
- Possibly the most important security-related ecommerce protocol
- Session sets up security parameters
- Many connections possible within a given session
- Current version TLS 1.0
  http://www.ietf.org/rfc/rfc2246.txt

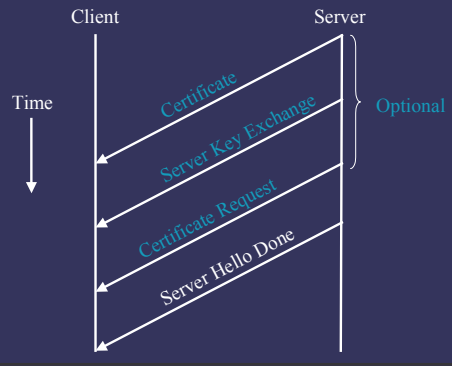
X.509 Key Est. Messages

- Let DA = EB(k), rA, LA, A.
- Let DB = rB, LB, rA, A
- Two messages:
  1. A -> B : certA, DA, SA(DA)
     Check that the nonce rA has not been seen, and is not expired according to LA. Remember it for its lifetime LA.
  2. B -> A : certB, DB, SB(DB)
     Check the rA and A. Check that rB has not been seen and is not expired according to LB.

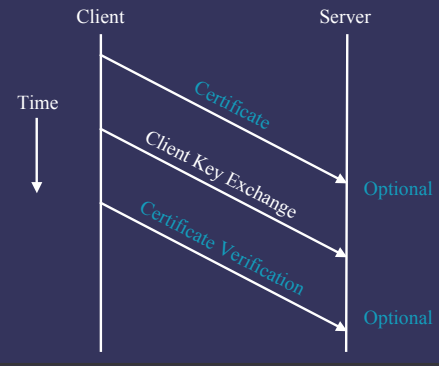
Establish Security Capabilities



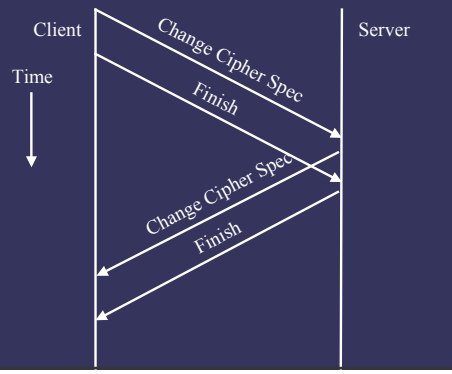
Server Auth & Key Exchange



Client Auth & Key Exchange



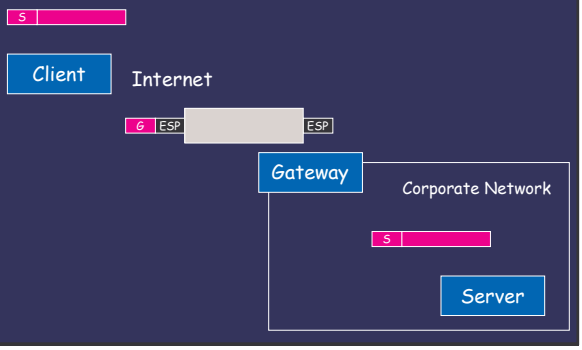
Client Auth & Key Exchange



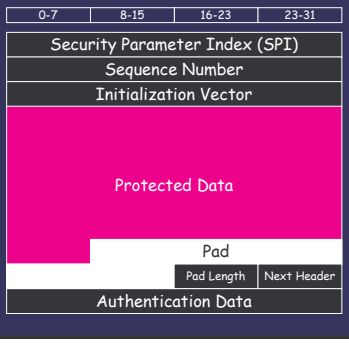
IPsec

- Modes
  - Tunnel
  - Transport
- Protocols
  - Authenticated Header (AH)
  - Encapsulated Security Payload (ESP)
- Configurations
  - End-to-end
  - Concatenated
  - Nested
- Principal elements
  - Security Associations (SAD)
  - Internet Key Exchange (IKE)
  - Policy (SPD)

Typical Case



Encapsulated Security Header and Trailer



Security Association

- An SA describes the parameters for processing a secured packet from one node to another
- SAs are simplex: use one for each direction
- If more than one SA is used for a packet the applicable SAs are called an "SA bundle"

SA Parameters (ESP Only)

- Sequence number, Sequence number overflow, Anti-replay window
- Lifetime
- Mode
- Tunnel destination
- PMTU
- Encryption algorithm (IV, etc.)
- Authentication algorithm

Policy

- Policy is not standardized in IPSec but certain basic functionality is expected
- A Security Policy Database (SPD) is consulted to determine what kind of security to apply to each packet
- The SPD is consulted during the processing of all traffic:
  - Inbound and outbound
  - IPSec and non-IPSec

SPD Actions

- Discard
- Bypass IPsec
- Apply IPsec: SPD must specify the security services to be provided.
  - For inbound traffic it is inferred from: destination address, protocol, SPI.
  - For outbound traffic this is done with a *selector*.

Selectors

- Selectors are predicates on packets that are used to map groups of packets to SAs or impose policy
- They are similar to firewall filters
- Selector support
  - Destination and source IP addresses
  - Name (DNS, X.509)
  - Source and destination ports (may not be available on inbound ESP packets; use inner header for inbound tunnel mode)

IPsec Processing: Outbound

- Use selectors in SPD to determine drop, bypass, or apply
- If apply, determine whether an SA or SA bundle for the packet exists
  - If yes, then apply all appropriate SAs before dispatching
  - If no, then create all necessary SAs. Apply these when done before dispatching

IPsec Processing: Inbound

- If there are no IPsec headers check SPD selectors to determine processing discard, bypass, or apply
- If apply, then drop
- If there are IPsec headers, apply SA determined by SPI, destination, protocol
- Use selectors on result to retrieve policy and confirm correct application

Internet Key Exchange (IKE)

- Motivating problem: Security settings (SAs) must be highly configurable
- Solutions:
  - Let network administrator manually configure SA (most common)
  - Provide mechanism to allow automatic negotiation and configuration
- Can be found at: http://ietf.org/internet-drafts/draft-ietf-ipsec-ikev2-13.txt
- IKEv2 Current as of March 22, 2004

Station to Station Protocol

1. A -> B : YA (Diffie-Hellman public key)
   Calculate k.
2. B -> A : YB, E(k, SB(YB, YA))
   Calculate k, use it to decrypt the signature, check the signature using the verification function of B and known values YB, YA.
3. A -> B : E(k, SA(YA, YB))
   Decrypt the signature and check it using the verification function of A.

High-level view

- Requester: Responder:
- IKE_SA_INIT -->
- <-- IKE_SA_INIT
- IKE_AUTH -->
- <-- IKE_AUTH
  - These are mandatory message exchange pairs, and must be executed in this order.

High-level view

- Initiator: Responder:
- CREATE_CHILD_SA -->
- <-- CREATE_CHILD_SA
- INFORMATIONAL -->
- <-- INFORMATIONAL
  - These messages are optional and can be sent by either party at any time.

Changes from IKEv1

- 4 initialization messages instead of 8
- Decrease latency in common case of 1 CHILD_SA by piggybacking this onto initial message exchanges
- Protocol is reliable (all messages are acknowledged and sequenced)
- Cookie exchange option ensures that the responder does not have to commit state until initiator proves it can accept messages

Summary

- PKI provides potential scalable identities for the Internet but adoption has been difficult
- Network protocols are designed in layers; security can be provided at multiple layers with various tradeoffs
- Theoretical protocols differ in significant ways from Internet standards and software