Internet Security Protocols

Bart Preneel
February 2007

With thanks to Joris Claessens and Walter Fumy
Outline

• Internet summary
• IETF process
• Basic principles
• Transport layer security
  – SSL / TLS
• Network layer security
  – IPSec, VPN, SSH
The Internet - A Network of Networks

• “IP is the protocol that integrates all infrastructures”
• **Network Layer**
  – Internet Protocol (IP)

• **Transport Layer**
  – Transmission Control Protocol (TCP), User Datagram Protocol (UDP)
Data Encapsulation

Application Layer (Web, FTP, ...)

Transport Layer (TCP, UDP)

Network Layer (IP)

Network Access Layer

IP Header

Transport Header

Data

Application

Transport

Network

Data

Transport Header

IP Header

Data

Application Layer (Web, FTP, ...)

Transport Layer (TCP, UDP)

Network Layer (IP)

Network Access Layer

Internet
Internet Standardization

- ISOC/IAB/IESG/IETF
- Internet Engineering Task Force (IETF)
- IETF Working Groups
  - Mailing List Information
  - Scope of the Working Group
  - Goals and Milestones
  - Current Internet Drafts & RFCs
- RFCs
  - http://www.rfc-editor.org
  - ftp://FTP.ISI.EDU/in-notes/
IETF Standards: RFC

- **Proposed Standard (PS)**
  - stable spec
  - lowest level of standards track

- **Draft Standard (DS)**
  - at least two independent and interoperable implementations

- **Standard (STD)**
  - widely, successfully used
IETF Intermediate documents

- Request for Comments (RFCs) with different maturity levels
  - Experimental (E)
  - Informational (I)
  - Historic (H)
  - Best Current Practice (BCP)

- **Internet-Drafts** (I-D) are working documents of the working groups and have no formal status

- Protocol Status (requirement level)
  - "required", "recommended", "elective",
    "limited use", or "not recommended"
  - “must” and “should”
IETF Security Area (1)

Area Directors: Russell Housley, Sam Hartman

- btns Better-Than-Nothing Security
- dkim Domain Keys Identified Mail
- emu EAP Method Update
- hokey Handover Keying
- idwg Intrusion Detection Exchange Format
- inch Extended Incident Handling
- isms Integrated Security Model for SNMP
- keyprov Provisioning of Symmetric Keys
- kink Kerberized Internet Negotiation of Keys
- kitten Kitten (GSS-API Next Generation)
- krb-wg Kerberos
- ltans Long-Term Archive and Notary Services
IETF Security Area (2)

*Area Directors: Russell Housley, Sam Hartman*

- mobie  IKEv2 Mobility and Multihoming
- msec  Multicast Security
- nea  Network Endpoint Assessment
- openpgp  An Open Specification for Pretty Good Privacy
- pki4ipsec  Profiling Use of PKI in IPSEC
- pkix  Public-Key Infrastructure (X.509)
- sasl  Simple Authentication and Security Layer
- secsh  Secure Shell
- smime  S/MIME Mail Security
- syslog  Security Issues in Network Event Logging
- Tls  Transport Layer Security
Communications insecurity

• architectural errors
  – wrong trust assumptions
  – default = no security

• protocol errors
  – unilateral entity authentication
  – weak entity authentication mechanism
  – downgrade attack

• modes of operation errors
  – no authenticated encryption
  – wrong use of crypto

• cryptographic errors
  – weak crypto

• implementation errors

range of wireless communication is often underestimated!
A historical perspective (1)

1900 wireless data
- Vernam: OTP
- rotor machines
- LFSR

1960

- WLAN
- PAN
- 3GSM

1980

1990

2000

1900 wired data
- block ciphers
- X25
- TLS
- SSH
- IPsec

1960

1980

1990

2000

1900 wired voice
- analog scramblers
- STU

1960

1980

1990

2000

digital encryption

VoIP
A historical perspective (2)

<table>
<thead>
<tr>
<th>Year</th>
<th>Mobile Phones</th>
<th>1990</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1980</td>
<td>AMPS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>analog cloning, scanners</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1990</td>
<td>GSM/TDMA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TDMA cloning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>3G GSM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>attacks on A5, COMP128</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>WLAN</th>
<th>1997</th>
<th>2002</th>
<th>2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>1997</td>
<td>WEP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2002</td>
<td>WPA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2004</td>
<td>WPA2 802.11i</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WEP broken</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WPA weak</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>PAN</th>
<th>1999</th>
<th>2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999</td>
<td>Bluetooth</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2004</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bluetooth problems</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Security Goals (started in ISO 7498-2)

• confidentiality:
  – entities (anonymity)
  – data
  – traffic flow

• (unilateral or mutual) entity authentication

• data authentication (connection-less or connection-oriented): data origin authentication + data integrity

• access control

• non-repudiation of origin versus deniability
Security Protocols & Services

• Cryptographic techniques:
  – symmetric encipherment
  – message authentication mechanisms
  – entity authentication mechanisms
  – key establishment mechanisms (e.g., combined with entity authentication)
Internet Security Protocols

- **Electronic Commerce Layer**
  - PayPal, Ecash, 3D Secure ...

- **Transport Layer Security** (SSH, SSL, TLS)

- **Transmission Control Protocol** (TCP)

- **User Datagram Protocol** (UDP)

- **IP/ IPSec (Internet Protocol Security)**

- **S-HTTP**

- **PGP**

- **PEM**

- **S/MIME**

- **PKIX**

- **SPKI**

- **Public-Key Infrastructure**

- **security services depend on the layer of integration:**
  - the mechanisms can only protect the payload and/or header information available at this layer
  - header information of lower layers is **not protected!!**
Security: at which layer?

- **Application layer:**
  - closer to user
  - more sophisticated/granular controls
  - end-to-end
  - but what about firewalls?

- **Lower layer:**
  - application independent
  - hide traffic data
  - but vulnerable in middle points

- **Combine?**
SP Architecture I: Encapsulation

- Bulk data: symmetric cryptography
- Authenticated encryption: best choice is to authenticate the ciphertext
SP Architecture II: Session (Association) Establishment

Security Associations
(Security Parameters incl. Shared Keys)

Key Management and Security Association Establishment Protocols
Algorithm Selection

"a la carte"

- each algorithm (encryption, integrity protection, pseudo-random function, Diffie-Hellman group, etc.) is negotiated independently
- less compact to encode
- more flexible

"suite"

- all parameters are encoded into a single suite number; negotiation consists of offering one or more suites and having the other side choose
- simpler and more compact to encode
- potentially exponential number of suites
- less flexible

- e.g., IKEv1
- e.g., TLS and IKEv2
Transport layer security

SSL / TLS
SSL/TLS Protocols

- connection-oriented data confidentiality and integrity, and optional client and server authentication.
Transport Layer Security Protocols

- IETF Working Group: *Transport Layer Security (tls)*
  - RFC 2246 (PS), 01/99
- transparent secure channels independent of the respective application.
- available protocols:
  - *Secure Shell (SSH)*, SSH Ltd.
  - *Secure Sockets Layer (SSL)*, Netscape
  - *Transport Layer Security (TLS)*, IETF
SSL / TLS

- Mainly in context of WWW security, i.e., to secure the HyperText Transfer Protocol (HTTP)
- But, in between application layer and TCP, thus can be used to secure other applications than HTTP too (IMAP, telnet, ftp, …)
Other WWW security protocols

- PCT: Microsoft’s alternative to SSL
- S-HTTP: S/MIME-like protocol
- SET: for credit card transactions
- XML-Signature: PKCS#7-based signature on XML documents
- ...

SSL / TLS

- “Secure Sockets Layer” (Netscape)
  - SSL 2.0: security flaws!
  - SSL 3.0: still widely used - not interoperable with TLS 1.0

- “Transport Layer Security” (IETF)
  - TLS 1.0: adopted SSL 3.0 with minor changes
  - RFC 2246, 01/99 (PS)

- TLS: security at the transport layer
  - can be used (and is intended) for other applications too
  - end-to-end secure channel, but nothing more...
  - data is only protected during communication
  - no non-repudiation!
Application
e.g., http, telnet, ...

Handshake Protocol

Client Hello
Server Hello
...

Change Cipher Spec
Protocol

Change Cipher Spec

Alert
Protocol

Application
Data

Alert

Application
Data

Record Layer Protocol

SSL record

Transport layer
TCP/IP
SSL/TLS in more detail

- “Record layer” protocol
  - fragmentation
  - compression (not in practice)
  - cryptographic security:
    - encryption $\rightarrow$ data confidentiality
    - MAC $\rightarrow$ data authentication [no digital signatures!]

- “Handshake” protocol
  - client and server authentication
  - establish cryptographic keys (for encryption and MAC)
  - negotiation of cryptographic algorithms
Handshake: overview

CLIENT

- Client Hello
- Certificate
- Client Key Exchange
- Certificate Verify
- [changecipherspec]
- Finished

SERVER

Hello Request

- Server Hello
- Certificate
- Server Key Exchange
- Certificate Request
- Server Hello Done
- [changecipherspec]
- Finished

✓ start handshake, protocol version, algorithms
✓ authentication server + exchange (pre)master secret
✓ client authentication
✓ end handshake, integrity verification
### TLS 1.0 Data Encapsulation Options

<table>
<thead>
<tr>
<th>Confidentiality</th>
<th>key size</th>
<th>algorithm options</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>40</td>
<td>RC4_40 RC2_CBC_40 DES_CBC_40</td>
</tr>
<tr>
<td></td>
<td>56</td>
<td>DES_CBC</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>RC4 IDEA_CBC</td>
</tr>
<tr>
<td></td>
<td>168</td>
<td>3DES_EDE_CBC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Integrity</th>
<th>key size</th>
<th>algorithm options</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>144</td>
<td>HMAC-MD5</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>HMAC-SHA</td>
</tr>
</tbody>
</table>
TLS 1.0 Key Management Options

Anonymous

Server authentication, no client authentication

DH_DSS
DH_RSA
DHE_DSS
DHE_RSA

Non anonymous

Server and client authentication

RSA
DH_DSS
DH_RSA
DHE_DSS
DHE_RSA

mandatory
### RFC 3268: AES Ciphersuites for TLS
06/2002

<table>
<thead>
<tr>
<th>CipherSuite</th>
<th>Key Exchange</th>
<th>Certificate Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLS_RSA_WITH_AES_128_CBC_SHA</td>
<td>RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DH_DSS_WITH_AES_128_CBC_SHA</td>
<td>DH_DSS</td>
<td>DSS</td>
</tr>
<tr>
<td>TLS_DH_RSA_WITH_AES_128_CBC_SHA</td>
<td>DH_RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DHE_DSS_WITH_AES_128_CBC_SHA</td>
<td>DHE_DSS</td>
<td>DSS</td>
</tr>
<tr>
<td>TLS_DHE_RSA_WITH_AES_128_CBC_SHA</td>
<td>DHE_RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DH_anon_WITH_AES_128_CBC_SHA</td>
<td>DH_anon</td>
<td></td>
</tr>
<tr>
<td>TLS_RSA_WITH_AES_256_CBC_SHA</td>
<td>RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DH_DSS_WITH_AES_256_CBC_SHA</td>
<td>DH_DSS</td>
<td>DSS</td>
</tr>
<tr>
<td>TLS_DH_RSA_WITH_AES_256_CBC_SHA</td>
<td>DH_RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DHE_DSS_WITH_AES_256_CBC_SHA</td>
<td>DHE_DSS</td>
<td>DSS</td>
</tr>
<tr>
<td>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</td>
<td>DHE_RSA</td>
<td>RSA</td>
</tr>
<tr>
<td>TLS_DH_anon_WITH_AES_256_CBC_SHA</td>
<td>DH_anon</td>
<td></td>
</tr>
</tbody>
</table>
TLS 1.1

- Makes RSA with 3DES the mandatory cipher suite (specifies no AES cipher suites - yet)
  - TLS 1.1: TLS_RSA_WITH_3DES_EDE_CBC_SHA
  - TLS 1.0: TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
- Provides several fixes, including
  - for the Rogaway and Vaudenay CBC attacks
  - for the Vaudenay, Boneh-Brumley, and KPR attacks

Version 1.2 will reduce dependency on MD5 and SHA-1
More IETF TLS

**Usage of TLS in HTTP:**
- upgrade to TLS within HTTP/1.1 (RFC 2817, 05/00)
- HTTP over TLS (RFC 2818, May 2000)

**Addition of ciphers:**
- Kerberos cipher suites (RFC 2712, 10/99; 11/00)
- ECC cipher suites (03/01)
- AES (01/01)
- misc. ciphers: MISTY1 (03/01), Camellia (10/00)
- extensions for OpenPGP keys (03/01)

**Other:**
- wireless extensions (11/00)
- TLS Delegation (02/01)
- SRP for TLS authentication (02/01)
TLS in the future (1)

• TLS 2.0 ?

• Some possible TLS enhancements, discussed within the IETF TLS WG:
  – RSA-OAEP
  – identity protection [note that this is already indirectly possible by authenticating within a DH_anon session]
  – cipher suites for compression
  – missing cipher suites (not all combinations possible)

• Backward compatibility remains very important!
TLS in the future (2)

TLS 1.1 – RFC June 2005
- security fixes and clarifications
- SSL/TLS is still in evolution!

Enhancements currently considered within IETF
- new cipher suites: e.g., AES, ECC
- wireless support (see WAP-WTLS) and other extensions
- password-based authentication and key exchange (SRP)

Other enhancements proposed in literature
- performance improvements:
  - ‘batching’ [ShachamBoneh’01] and ‘fast-track’ [ShachamBoneh’02]
- user (identity) privacy [PersianoVisconti’00]
- client puzzles [DeanStubblefield’01] to counter denial-of-service attacks
- trust negotiation [Hess et al’02]
SSL/TLS: security services

SSL/TLS only provides:
• entity authentication
• data confidentiality
• data authentication

SSL/TLS does not provide:
• non-repudiation
• unobservability (identity privacy)
• protection against traffic analysis
• secure many-to-many communications (multicast)
• security of the end-points (but relies on it!)
SSL/TLS: security?

- TLS 1.0 is the result of a public reviewing process: several problems have been identified in earlier versions (SSL 2.0/3.0) and have been solved
- SSL/TLS is practically secure
- Some caveats (in order of importance):
  - bad implementation; e.g., random number generation
  - PKCS#1 attack (use other padding scheme: OAEP; server error messages should contain less information)
  - version / cipher suite roll back attempts (due to backward compatibility; disable export algorithms if possible)
  - traffic analysis: e.g., length of ciphertext might reveal useful info
  - plenty of known plaintext (both SSL/TLS and HTTP related)
SSL/TLS: evaluation

TLS 1.0 provides a good level of security

– result of a public reviewing process: several problems have been identified in earlier versions (SSL 2.0/3.0) and have been addressed

Some remaining security problems though

– downgrade attacks
– cryptographic attacks
– PKI related problems
– web spoofing
– platform and users
Security in transport layer

• Transparent for application
• Pro: can be used for all TCP-based applications, without modifying them
• Con: authentication is one, but who/what to trust, is important
• Non-repudiation?
• In practice: (partially) integrated in application
Non-repudiation

• Legally only if in application, thus not provided by SSL/TLS
• SSL/TLS secures the communication channel, but not the exchanged messages
• SSL/TLS does not use digital signatures in the first place (except for client authentication)
• For electronic business, more advanced security protocols are needed...
User authentication

First authentication, then authorization!

SSL/TLS client authentication:
- during handshake, client digitally signs a specific message that depends on all relevant parameters of secure session with server
- software devices, smart cards or USB tokens can be deployed through standardized cryptographic interfaces supported by browsers
  (Netscape: PKCS#11; MSIE: PC/SC)
- PKCS#12 key container provides software mobility

Usually another mechanism on top of SSL/TLS
Network layer security

IPsec, VPN, SSH
IP Security Protocols

- IETF Working Group:  
  **IP Security Protocol (ipsec)**  
  Security Architecture for the Internet Protocol  
  - RFC 2401 (PS), 11/98
- **IP Authentication Header (AH)**  
  - RFC 2402 (PS), 11/98
- **IP Encapsulating Security Payload (ESP)**  
  - RFC 2406 (PS), 11/98
- **Internet Key Exchange (IKE)**  
  - RFC 2409 (PS), 11/98  
  - Application layer protocol for negotiation of Security Associations (SA) and Key Establishment

- Large and complex……………. (48 documents)
- Mandatory for IPv6, optional for IPv4
IPSec VPN models:
Hosts and Security Gateways

Host-to-host (not VPN)

Branch-to-branch

Host-to-gateway
IPsec - Security services

- Access control
- Connectionless integrity
- Data origin authentication
- Rejection of replayed packets (a form of partial sequence integrity)
- Confidentiality
- Limited traffic flow confidentiality
IPsec - Concepts

• Security features are added as extension headers that follow the main IP header
  – Authentication header (AH)
  – Encapsulating Security Payload (ESP) header

• Security Association (SA)
  – Security Parameter Index (SPI)
  – IP destination address
  – Security Protocol Identifier (AH or ESP)
IPsec - Parameters

- sequence number counter
- sequence counter overflow
- anti-replay window
- AH info (algorithm, keys, lifetimes, ...)
- ESP info (algorithms, keys, IVs, lifetimes, ...)
- lifetime
- IPSec protocol mode (tunnel or transport)
- path MTU (maximum transmission unit)
# IKE Algorithm Selection

<table>
<thead>
<tr>
<th>Algorithm Type</th>
<th>IKE v1</th>
<th>IKE v2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payload Encryption</td>
<td>DES-CBC</td>
<td>AES-128-CBC</td>
</tr>
<tr>
<td>Payload Integrity</td>
<td>HMAC-MD5, HMAC-SHA1</td>
<td>HMAC-SHA1</td>
</tr>
<tr>
<td>DH Group</td>
<td>768 Bit</td>
<td>1536 Bit</td>
</tr>
<tr>
<td>Transfer Type 1</td>
<td>ENCR_DES_CBC</td>
<td>ENCR_AES_128_CBC</td>
</tr>
<tr>
<td>(Encryption)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transfer Type 2</td>
<td>PRF_HMAC_SHA1 [RFC2104]</td>
<td>PRF_HMAC_SHA1 [RFC2104]</td>
</tr>
<tr>
<td>(PRF)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transfer Type 3</td>
<td>AUTH_HMAC_SHA1_96 [RFC2404]</td>
<td>AUTH_HMAC_SHA1_96 [RFC2404]</td>
</tr>
<tr>
<td>(Integrity)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: draft-ietf-ipsec-ikev2-algorithms-00.txt, May 2003
IPsec - Modes

• Transport (*host-to-host*)
  – ESP: encrypts and optionally authenticates IP payload, but not IP header
  – AH: authenticates IP payload and selected portions of IP header

• Tunnel (*between security gateways*)
  – after AH or ESP fields are added, the entire packet is treated as payload of new outer IP packet with new outer header
  – used for VPN
IPsec - AH Transport mode

- Security Parameters Index: identifies SA
- Sequence number: anti-replay
- Integrity Check Value: data authentication using HMAC-SHA-1-96 or HMAC-MD5-96

*Integrity*

(only header fields that are not changed or are changed in a predictable manner)
IPsec - AH Tunnel mode

**Integrity**
*only header fields that are not changed or are changed in a predictable manner*
IPsec - ESP header

- Security Parameters Index: identifies SA
- Sequence number: anti-replay
- Encrypted payload data: data confidentiality using DES, 3DES, RC5, IDEA, CAST, Blowfish
- Padding: required by encryption algorithm (additional padding to provide traffic flow confidentiality)
- Integrity Check Value: data authentication using HMAC-SHA-1-96 or HMAC-MD5-96
IPsec - ESP Transport mode
IPsec - ESP Tunnel mode

IP hdr  upper layer data

new IP hdr  ESP hdr  IP hdr  upper layer data  ESP tlr  ICV

Confidentiality

Integrity
IPsec - Key management

- RFCs 2407, 2408, and 2409
- Manual
- Automated
  - procedure / framework
    - Internet Security Association and Key Management Protocol (ISAKMP), RFC 2408 (PS)
  - key exchange mechanism: Internet Key Exchange (IKE)
    - Oakley: DH + cookie mechanism to thwart clogging attacks
    - SKEME
IPsec: Key management

• IKE defines 5 exchanges
  – Phase 1: establish a secure channel
    • Main mode
    • Aggressive mode
  – Phase 2: negotiate IPSEC security association
    • Quick mode (only hashes, PRFs)
  – Informational exchanges: status, new DH group

• based on 5 generic exchanges defined in ISAKMP

• cookies for anti-clogging
IPsec: Key management

• protection suite (negotiated)
  – encryption algorithm
  – hash algorithm
  – authentication method:
    • preshared keys, DSA, RSA, encrypted nonces
  – Diffie Hellman group: 5 possibilities
IKE - Main Mode with Digital Signatures

**Initiator**

- **K** derived from
  
  \[
  \text{master} = \text{prf} (N_i \parallel N_r, g^{xy})
  \]

- \(\text{SIG}_i = \text{Signature on}\)

  \(H(\text{master}, g^x \parallel g^y \parallel ... \parallel ID_i)\)

**Responder**

- **SIG** \(_r\) = Signature on

  \(H(\text{master}, g^y \parallel g^x \parallel ... \parallel ID_r)\)

- \(E(K, ID_i, [\text{Cert}(i)], \text{SIG}_i)\)

- \(E(K, ID_r, [\text{Cert}(r)], \text{SIG}_r)\)

\(H\) is equal to prf or the hash function tied to the signature algorithm
(all inputs are concatenated)
IKE - Main Mode with Digital Signatures

- mutual entity authentication
- mutual implicit and explicit key authentication
- mutual key confirmation
- joint key control
- identity protection
- freshness of keying material
- perfect forward secrecy of keying material
- non-repudiation of communication
- cryptographic algorithm negotiation
IKE v2 - RFC Dec 2005

• IKEv1 implementations incorporate additional functionality including features for NAT traversal, legacy authentication, and remote address acquisition, not documented in the base documents.

• Goals of the IKEv2 specification include
  – to specify all that functionality in a single document
  – to simplify and improve the protocol, and to fix various problems in IKEv1 that had been found through deployment or analysis

• IKEv2 preserves most of the IKEv1 features while redesigning the protocol for efficiency, security, robustness, and flexibility.
IKE v2 Initial Handshake (1/2)

• Alice and Bob negotiate cryptographic algorithms, mutually authenticate, and establish a session key, creating an IKE-SA

• Usually consists of two request/response pairs
  – The first pair negotiates cryptographic algorithms and does a Diffie-Hellman exchange
  – The second pair is encrypted and integrity protected with keys based on the Diffie-Hellman exchange
IKE v2 Initial Handshake (2/2)

• Second exchange
  – divulge identities
  – prove identities using an integrity check based on the secret associated with their identity (private key or shared secret key) and the contents of the first pair of messages in the exchange
  – establish a first IPsec SA (“child-SA”) is during the initial IKE-SA creation
IPsec Overview

• Much better than previous alternatives
• IPsec documents hard to read
• Committee design: too complex
  – ESP in Tunnel mode probably sufficient
  – Simplify key management
  – Clarify cryptographic requirements
• …and thus difficult to implement (securely)
VPN?

• **Virtual Private Network**
• Connects a private network over a public network.
• Connection is secured by tunneling protocols.
• The nature of the public network is irrelevant to the user.
• It appears as if the data is being sent over the private network.
Transit Internetwork

Virtual Private Network

Logical Equivalent
VPN - Common use

• Remote user access over the Internet

• Connecting networks over the Internet

• Connection computers over an intranet
Remote user access over the Internet

- You can use existing local Internet connections.
- No need for long distance connections
Connecting networks over the Internet

- You can use existing local Internet connections.
- No need for long distance connections or leased lines
Connecting computers over an intranet

- Provides easy client access to secured or hidden networks within the corporate network
VPN - Basic requirements

• User authentication and user authorization
• Data authentication and data confidentiality
• Key management
• Encapsulation
  – data of private network is encapsulated in packets suited for transmission over the public network. (tunneling protocol)
• Address management
  – assign a client’s address on the private net
Tunneling
Final remarks
Some observations

• IPSec is really transparent, SSL/TLS only conceptually, but not really in practice
• SSH, PGP: stand-alone applications, immediately and easy to deploy and use
• Network security: solved in principle
• Electronic commerce security: more is needed!
More information (1)


• IETF web site: www.ietf.org
  – e.g., IETF-TLS Working Group
    http://www.ietf.org/html.charters/tls-charter.html
More information (2)

- **Java Security (2nd edition)**
  http://www.securingjava.com/
- **W3C Security (incl WWW Security FAQ)**
  http://www.w3.org/Security/
- **“E-Commerce Security, Weak Links, Best Defenses”**
  http://www.cigital.com/books/ecs/
- **“Security Technologies for the World Wide Web”**
  http://www.esecurity.ch/Books/wwwsec.html