Practical Aspects of Modern Cryptography

Josh Benaloh & Brian LaMacchia

Lecture 10: IPSEC and Crypto Politics

But First...We Need to Vote

- Two choices for the final project schedule:
- Option 1:
  - Tuesday, March 19 @ UW EE1-003
  - Thursday, March 21 @ Redmond (room TBD)
- Option 2:
  - Tuesday, March 19 @ Redmond (room TBD)
  - Thursday, March 21 @ UW EE1-003
- Which do you prefer?

IPSEC

- IPSEC = IP (Internet Protocol) Security
  - Suite of protocols that provide encryption, integrity and authentication services for IP packets
  - Mandatory-to-implement for IPv6, optional (but available) for IPv4
- Consists of two main components:
  - IPSEC proper (encryption & auth of IP packets)
  - IPSEC key management

IPSEC Operation

- Provides two modes of protection
  - Tunnel Mode
  - Transport Mode
- Protection protocols
  - Authentication and Integrity (AH)
  - Confidentiality (ESP)
  - Replay Protection

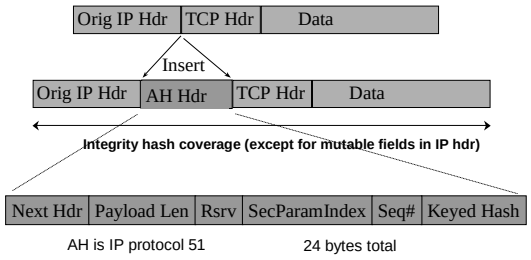
IPSEC Protection Protocols

- Authentication Header (AH)
  - Authenticates payload data
  - Authenticates network header
  - Gives anti-replay protection
- Encapsulated Security Payload (ESP)
  - Encrypts payload data
  - Authenticates payload data
  - Gives anti-replay protection

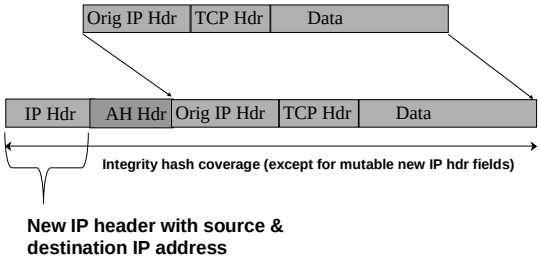
Authentication Header (AH)

- Authentication is applied to the entire packet, with the mutable fields in the IP header zeroed out
- If both ESP and AH are applied to a packet, AH follows ESP

IPSEC Authentication Header (AH) in Transport Mode



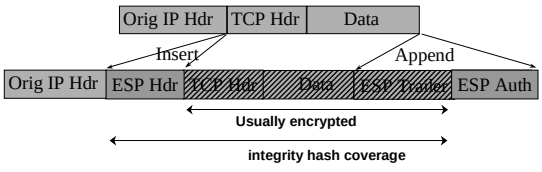
IPSEC AH in Tunnel Mode



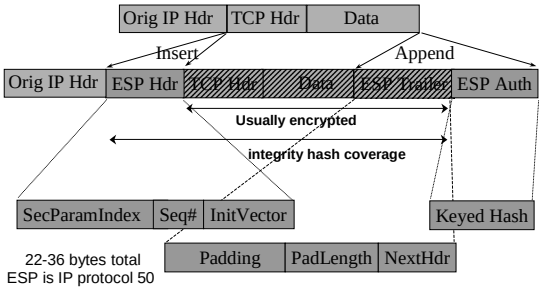
Encapsulated Security Payload (ESP)

- Must encrypt and/or authenticate in each packet
- Encryption occurs before authentication
- Authentication is applied to data in the IPSEC header as well as the data contained as payload

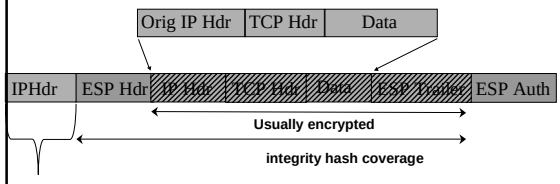
IPSEC ESP in Transport Mode



IPSEC ESP in Transport Mode



IPSEC ESP Tunnel Mode



New IP header with source & destination IP address

IPSEC Key Management

- IPSEC Key Management is all about establishing and maintaining Security Associations (SAs) between pairs of communicating hosts

Security Associations (SA)

- New concept for IP communication
  - SA not a "connection", but very similar
  - Establishes trust between computers
- If securing with IPSEC, need SA
  - ISAKMP protocol negotiates security parameters according to policy
  - Manages cryptographic keys and lifetime
  - Enforces trust by mutual authentication

Internet Key Exchange (IKE)

- Phase I
  - Establish a secure channel(ISAKMP SA)
  - Authenticate computer identity
- Phase II
  - Establishes a secure channel between computers intended for the transmission of data (IPSEC SA)

ISAKMP/OAKLEY

- Merge of two key management protocols
  - ISAKMP: Internet Security Association and Key Management Protocol
    - NSA-designed protocol to exchange security parameters (but not establish keys)
  - OAKLEY
    - Diffie-Hellman based key management protocol

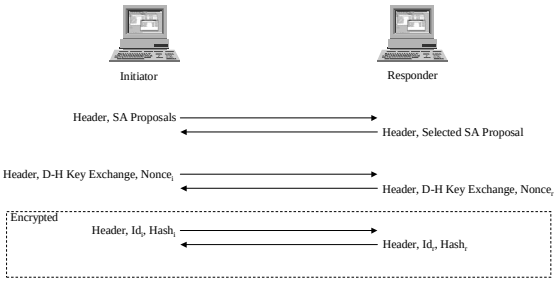
ISAKMP/OAKLEY (2)

- What's used today is a combination
  - ISAKMP provides the protocol framework
  - OAKLEY provides the security mechanisms

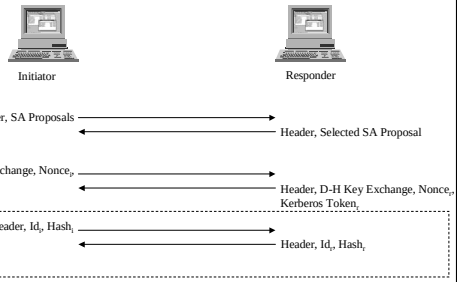
Main Mode

- Main mode negotiates an ISAKMP SA which will be used to create IPSEC SA
- Three steps
  - SA negotiation
  - Diffie-Hellman and nonce exchange
  - Authentication

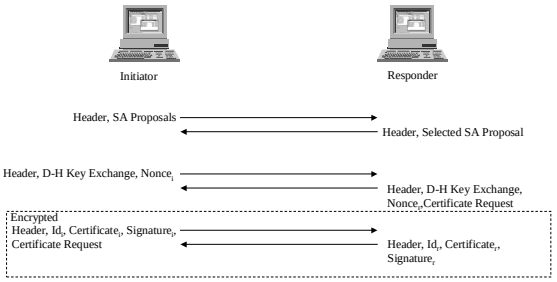
Main Mode (Pre-shared Key)



Main Mode (Kerberos)



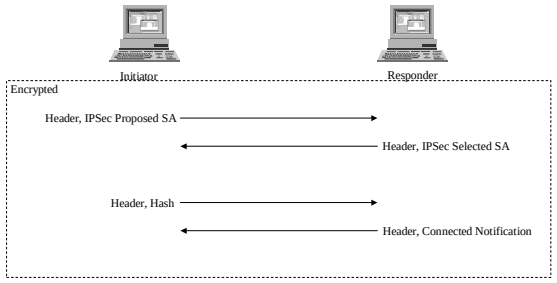
Main Mode (Certificate)



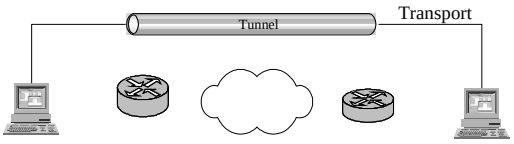
Quick Mode

- All traffic is encrypted using the ISAKMP Security Association
- Each quick mode negotiation results in two IPSec Security Associations (one inbound, one outbound)

Quick Mode Negotiation



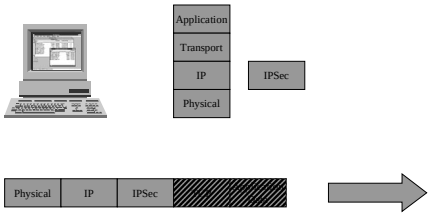
How It All Fits Together



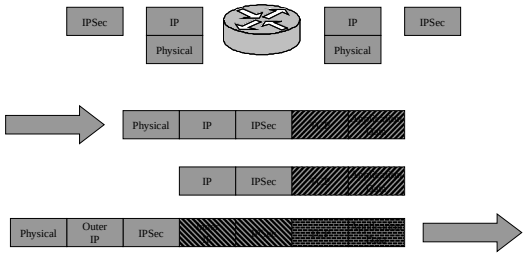
IPSEC Bundling/Wrapping

- Multiple IPSEC transforms may be wrapped successively around a single IP datagram
  - Example: IPSEC transport sent over an IPSEC tunnel

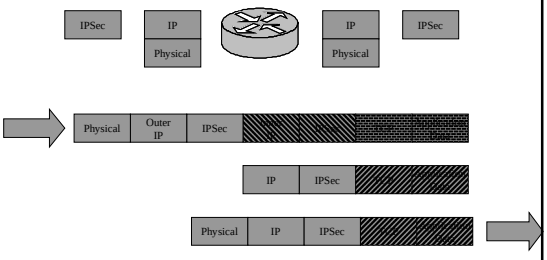
Sending in Transport Mode



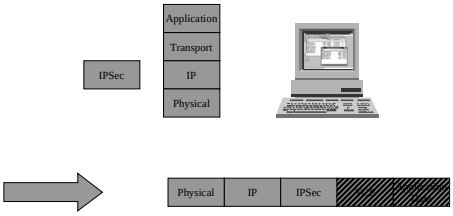
Sending in Tunnel Mode



Receiving in Tunnel Mode



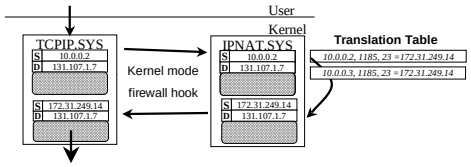
Receiving in Transport Mode



What is Network Address Translation (NAT) ?

- Network Address Translation (NAT)
  - Dynamically modifies source address
  - Dynamically recomputes interior UDP/TCP checksums
- Port Address Translation (PAT)
  - Dynamically modifies TCP/UDP source address and port
  - Dynamically recomputes interior UDP/TCP checksums

NATs Rewrite Address/Port Pairs

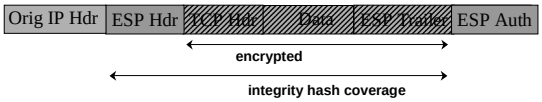


IPSEC AH and NAT

- Change in address or port will cause message integrity check to fail
  - Packet will be rejected by destination IPSEC
  - AH cannot be used with NAT or PAT devices

IPSEC ESP and NAT

- Can change IP header in special cases only
  - Special TCP/UDP ignores pseudo header used in checksum calculation
- Port information encrypted!
- Can't change ESP header because integrity hash coverage



The Politics of Crypto

Why Talk About Crypto Politics?

- You can't really avoid the political aspects of crypto, especially if you're trying to ship a product that depends on good crypto
  - In the past, the regulations have been so complex & time consuming that companies had dedicated individuals/departments for dealing with regs.
- Often public pronouncements don't match reality
  - Just because a government body says "crypto is freely exportable" doesn't make it so

Topics in Crypto Politics

- Export controls
- Key Escrow
- Patents
- Copyright

Caveats...

- I'm going to present a U.S.-centric view of the issues
  - Each country deals differently with these issues, but the U.S. typically leads in this policy area
- These are *national issues* – nation-states are still important to the discussion
- Much of what we have learned about the history of export controls has come from FOIA requests
  - The government doesn't like to give answers...

Export Controls

- Question 1:

  "Should the export of cryptographic software from the U.S. be restricted? If not, why not? If so, why and to what degree?"

  Discuss.

Export Controls

"Should the export of cryptographic software from the U.S. be restricted? If not, why not? If so, why and to what degree?"

- For the next 2-3 minutes, think about how you would individually respond to this question.
  - You might find it helpful to organize your thoughts into "pros" and "cons"
  - Just brainstorm for the next few minutes...

Export Controls

- Question 1:

  "Should the export of cryptographic software from the U.S. be restricted? If not, why not? If so, why and to what degree?"

  Discuss.

Export Controls

- OK, now that you've thought about it a bit, talk with your neighbors and see what they came up with...

Export Controls

- Question 1:

  "Should the export of cryptographic software from the U.S. be restricted? If not, why not? If so, why and to what degree?"

  Discuss.

Export Controls in the U.S.

- In the beginning, cryptographic hardware and software were considered "munitions" by the U.S. government.
  - Export of crypto was covered by the same set of regulations that covered the export of other munitions, like nuclear weapons, missiles, and the equipment that is used to make them
  - These regulations were known as ITAR (International Traffic in Arms Regulations).

Export Controls (cont.)

- Under ITAR, all exports of crypto required a license
  - If you were exporting "weak crypto" you could get a license.
    - "Strong crypto" couldn't be exported at all.
    - "Crypto with a hole" couldn't be exported either.
  - The distinction between "weak" and "strong" was generally based on bit-length of the secret key or public key modulus

Crypto Export/Import Controls

- The export of cryptography is currently restricted by the U.S. BXA (Commerce Dept. Bureau of Export Administration)
  - Until January 2000, couldn't export symmetric ciphers using keys > 56 bits in length.
  - Jan 2000: Clinton administration rewrote the regulations
    - "ITAR" became "EAR", and the regulations got a bit "looser" but they still exist
    - You can (generally speaking) export "strong crypto" without a specific product license

Current Export Regulations

- "Monolithic applications" can export strong cryptography in binary form simply by sending the BXA a piece of e-mail
  - Example: secure e-mail client, web browser
- "Crypto libraries" can be exported under an "open source" exemption, if they qualify
- "Crypto with a hole" in commercial products is still tightly controlled

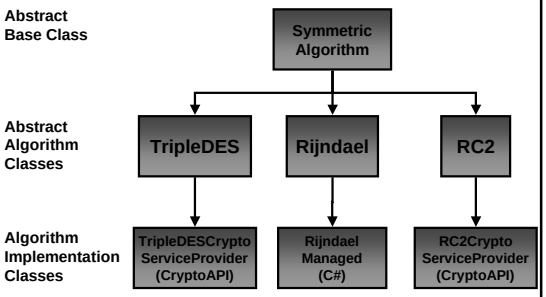
Example: Windows XP

- Windows XP ships with "strong crypto" baked in & enabled
  - RSA to 4096 bits, TripleDES, etc.
- Windows XP is exportable because it's a "monolithic application"
- CryptoAPI, the Win32 crypto library that was designed to support plug-able "cryptographic service providers" is **not** freely exportable
  - If you want to plug into CryptoAPI, you need a license...

The Regs are Still Ambiguous

- In the .NET Framework, we have a class library for cryptography…
- It took BXA (really, NSA) 18 months to tell us what the rules were regarding export of our class library…

.NET FX Crypto Object Model



The Regs are Still Ambiguous

- In the .NET Framework, we have a class library for cryptography…
- It took BXA (really, NSA) 18 months to tell us what the rules were regarding export of our class library…
- We could open up & let people subclass the bottom abstract classes (like RSA) without a license
- Opening up AsymmetricAlgorithm was not allowed without an explicit license
- Solution? Open source the code!

More Export Control Stories

- Bruce Schneier and the first edition of *Applied Cryptography*
  - Phil Karn's attempt
  - Dan Bernstein's attempt
    - http://www.eff.org/bernstein/
- Matt Blaze and the "fancy gun"
  - http://www.frogtown.com/pipermail/funny/1996-January/000150.html
- Phil Zimmermann and PGP 1.0

Key Escrow

- The general topic of "key escrow" is about archiving copies of private keys with third parties.
  - This is also sometimes called "key archival"
  - When the government is the archive, this is GAK (Government Access to Keys)
- There are legitimate cases where you might need a key escrow scheme
  - Stored data recovery in case of accident/loss/termination of employment

Key Escrow

- There are no legitimate cases (at least from a commercial perspective) for archival of secret session keys.
  - If the data didn't get transmitted correctly during the session, send it again
- Governments care about session encryption key recovery
  - Want to preserve their wiretapping capabilities
- Government spent a lot of time trying to convince businesses that the needs of stored data recovery & session key recovery were the same

Key Escrow

- Question 2:

  "Should a national government have the right to demand encryption keys from citizens when (a) they are suspects in a criminal investigation, or (b) in cases of `national security'? If not, why not? If so, what procedures should the government have to follow to obtain the keys?"

  "Have your feelings changed post-9/11?"

  Discuss.

Key Escrow

- Again, brainstorm on your own for 2-3 minutes, then discuss for a few minutes with your neighbors.

Key Escrow

- Question 2:

  "Should a national government have the right to demand encryption keys from citizens when (a) they are suspects in a criminal investigation, or (b) in cases of `national security'? If not, why not? If so, what procedures should the government have to follow to obtain the keys?"

  "Have your feelings changed post-9/11?"

  Discuss.

Digital Telephony

- In the U.S., the digitization of the nation's telephone system was seen by law enforcement as a threat to their ability to conduct wiretaps
  - In the analog world, you just go tap a pair of wires
  - In the digital world, you need to sift out the right bits from the optical fiber.
    - Even if you find the bits, they could be encrypted!

Digital Telephony & CALEA

- U.S. Congress response to law enforcement was to pass laws mandating that telephone companies guarantee wiretap access to their customers' communications
  - Communications Assistance for Law Enforcement Act (CALEA), Oct. '94
    - FBI said $150M-$500M in '92-94
    - Industry said cost would be $3B in '94
    - As of '98, est. $8B/year, $12M per wiretap
- CALEA still isn't a reality (cost, tech difficulty)
- CALEA doesn't help if the bits themselves are encrypted! FBI needed something else…

The Clipper Chip

The Clipper Chip

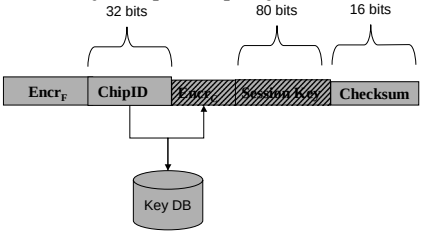
- US Government attempt to "stimulate" the market for "voluntary" key escrow equipment
  - Contracted w/ AT&T to produce "Clipper phones" for government use
  - Phones would also be available for non-government use
  - Encryption keys could be accessed through the "Law Enforcement Access Field" (LEAF) in the protocol

How Clipper Worked

- Clipper was implemented in a tamper-resistant hardware device (a single chip)
  - Each chip was numbered and had a separate per-chip secret that was also held by a "trusted agency" (read: US Gov't)
- Per-session keys were encrypted with a Clipper family key and the per-chip key, and sent along as part of the data stream
- Someone listening in on the conversation would see enough information to identify the chip used to encrypt, find the per-chip key, and recover the session key

How Clipper Worked (2)

- 128-bit LEAF contains session key encrypted with family and per-chip keys



Clipper in Operation

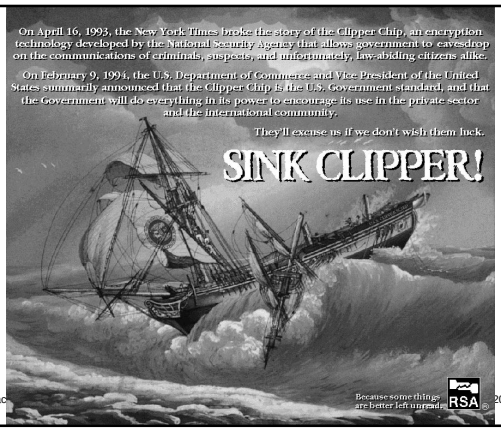
- Other party & third-party decrypt LEAF with the family key
- Both parties check the checksum to detect bogus LEAF
  - Bogus LEAF → chip turns off, refuses to decrypt
- Third party looks up chip key in DB to decrypt session key

Clipper Weaknesses

- The 80-bit session key was too small
- The symmetric cipher (SKIPJACK) was classified; no public scrutiny
  - Later, a "panel of outside experts" was allowed to look at it for a day
  - Even later, after Clipper failed, SKIPJACK was declassified
- 16-bit checksum could be defeated (Blaze '94)
- ChipID tagged every single communication

Opposition to Clipper

- Opposition to Clipper was widespread
  - The US Gov't proposed it as the federal Escrowed Encryption Standard and rammed it through NIST into FIPS 185 in Feb '94
  - During the public comment period, 300 comments received, only 2 supported it
- No one bought Clipper
  - AT&T shut down its product line, offered leftover phones to employees to get rid of them
- Oddly, the proposal probably did more to galvanize the strong-crypto community than anything else



Patents

- Crypto has a long, involved history with the US Patent Office
- The RSA patent was one of the first (if not *the* first) "algorithm" patent
  - You can't actually patent an algorithm, so RSA patented every type of machine/embodiment that implemented the algorithm
  - For 17 years, RSA was patented in the US, but freely available overseas

Copyright

- More recently, cryptography has become an issue in the area of *copyright*.
- Why?
- The rise of digital rights management (DRM) systems, all of which are based on strong crypto.
  - Break the crypto, break the DRM…

Copyright & DRM

- Digital Rights Management (DRM) technologies limit access to digital intellectual property.
  - Example: A DRM-protected e-book might let you read the book only a fixed number of times.
  - Example: A DRM-protected streaming audio player could charge you based on bandwidth & content.
- Major issues:
  - How restrictive can a DRM be?
  - How restrictive **should** a DRM be?
  - How do DRMs interact with "fair use" and other copyright rights reserved to the public?

Digital Millennium Copyright Act (DMCA)

- Characterized by proponents as a "small, technical" change to US copyright law
  - In reality, made major, sweeping provisions to the rules regarding digital content
- Incorporated into U.S. law at 17 USC 1201 *et. sec.*
  - "No person shall circumvent a technological measure that effectively controls access to a work protected under [copyright]…"

Anti-Circumvention Measures

- The DMCA made it a crime to circumvent a "technological measure that effectively controls access to a work"
  - "A technological measure 'effectively controls access to a work' if the measure, in the ordinary course of its operation, requires the application of information…with the authority of the copyright owner, to gain access to the work.
- Limited exemptions for
  - Encryption research
  - Reverse-engineering computer programs for interoperability.

DMCA cases/issues (1)

- DeCSS
  - DVDs are encrypted. In order to play a DVD, a licensed DVD play must first authenticate to the DVD disk.
  - DeCSS is a program that removes/bypasses the encryption, allowing the DVD to be played on an "unlicensed" player, such as a Linux box.
  - MPAA sued, claiming DCMA violations
  - Upheld in NY

DMCA cases/issues (2)

- CPHack
  - CyberPatrol (owned by Mattel, Inc.) is your typical "parental filter" for web browsers
  - CP's list of banned web sites is encrypted (using a secret algorithm) as part of the program
  - Jansson & Skala figured out how to break the encryption scheme (in a very nice piece of cryptanalysis)
  - They write a program, CPHack, which shows you the list of banned sites on your copy of CP.
  - Mattel sues

DMCA cases/issues (3)

- SDMI (*Felten v. RIAA*)
  - DMCA was used to threaten an academic group that successfully broke a number of proposed watermarking technologies (at least one of which is being used in commercial product)

DMCA cases/issues (4)

- bnetd
  - Blizzard Entertainment (a game software company) puts out a beta test of Warcraft III
  - W3 talks to Blizzard servers for on-line multiplayer games
    - The Blizzard servers authenticate software copies when the user logs in (looking for infringing copies...)
  - The servers were slow, so some enterprising folks reverse-engineered the protocol and started running their own servers (bnetd servers)

DMCA cases/issues (4 cont)

- Blizzard threatened the authors of bnetd with a DMCA complaint
  - They claimed that because bnetd doesn't implement the CD auth protocol, it facilitates infringement
  - bnetd folks pulled the software