CSE 461 – Network Security

David Wetherall
djw@cs.washington.edu
Network Security

• Focus
  – How do we secure network systems?

• Topics
  – A brief tour of the landscape
  – If you like it, take a security course (484)!
Overall picture

• Security is a generic term, like “performance”
  – Need to know what you’re trying to stop (threat / attack model)
  – Reduce to what you want system to ensure (security properties)

• Security is hard
  – It’s a negative goal, undone by any weakness (design flaw, bug, users)
  – Real security is risk management, not mathematics

• The balance today
  – Cryptography is a powerful, principled set of tools at our disposal
  – Exploits are design flaws (use crypto the wrong way), implementation
    bugs (buffer overruns), and usage failures (social engineering)
Threat Models

- We might be concerned with any of these:
  - Eavesdropping (Eve hears what Alice/Bob say)
  - Tampering (Trudy alters Alice’s messages to Bob)
  - Impersonation (Trudy pretends to be Alice to Bob)
  - Denial of service (Trudy stops Alice and Bob from talking)
  - ...

- i.e., more than encrypting messages ("classic security")
Security Properties

- Might want any/all of these properties
  - Privacy: messages can’t be eavesdropped
  - Integrity: messages can’t be tampered with
  - Authenticity: we can verify who created the message
  - Timeliness: we can verify that the packet was sent not too long ago
  - Availability: I can send and receive the packets I want
  - Non-repudiation: you can’t claim you didn’t say something you did
  - Anonymity: not only can’t you tell what the content of my conversation is, you can’t even tell who I’m talking with

- There are other properties we would like from the distributed services that run on top, as well
  - E.g., if I send you my medical records, you can’t send them to anyone else
Classic encryption

• Will assume it works; don’t have time to delve into it

• Can use encryption to ensure that messages can’t be read or tampered with by others, and come from a legitimate sender
  – Often called confidentiality/secrecy, integrity/authenticity

• Secret key encryption (e.g., AES)
• Public key encryption (e.g., RSA)
• Message digests (cryptographic hashes, e.g., SHA-1)
Secret Key Encryption (AES, 3DES)

- Single key (symmetric) is a secret shared between parties
  - Used both for encryption and decryption
  - Algorithm mangles bits as parameterized by the key

- Pros: Fast
- Cons: Need to distribute the secret keys!
Example issue: Repeated Messages

• Algorithms encrypt a fixed size block
  – Think of them as a 1-1 mapping of blocks of bits

• What happens when the message has repeated blocks?
  – Obvious solution will give the same encrypted block (for same key)
  – But this leaks information to eavesdropper!

• Called Electronic Code Book mode (ECB)
  – Has other problems too (no integrity for long messages)
  – Instead, we use other modes (that initialize and mix blocks)
Public Key Encryption (RSA)

- Key is split into public and private parts
  - Public key can be published; used to encrypt messages
  - Private is a secret; used to decrypt messages
- Pros: Only need to distribution public keys (one per user)
- Cons: Slow (say 1000X secret key)
Cryptographic Hash / Message Digest

• Basically:
  – Mangles message bits to a fixed size result, like a checksum
    • Flipping any message bit should randomly flip the digest bits
  – Difficult (infeasible) to find two messages with the same one
    • Not a 1-1 mapping, but digest size is too large to search

• Significance:
  – Digest can “stand in” for a long message, e.g., encrypt digest

• Examples: SHA-1 (160 bits), MD5 (now broken)
Network security

1. Security at different layers
   - Link, network, transport, application, …

2. Usage vulnerabilities
   - Social engineering

3. Application vulnerabilities
   - Buffer overruns, SQL injection …

4. Security at administrative boundaries
   - firewalls, ISPs, …

5. Co-opting or abusing network protocols
   - DDOS floods, DNS poisoning, TCP SYN floods, …
1. Application layer: ssh

• Secure (remote login) shell between two computers
  – Protects messages with confidentiality, integrity / authenticity

• Initially uses public key encryption to verify remote party
  – Q: how is public key distributed? who is verifying who?

• Session key:
  – Common practice to use public keys to authenticate initial contact, then switch to random agreed secret keys for better performance
  – Secret key is ephemeral and called a session key
Transport layer: SSL

- HTTPS is HTTP on top of Secure Socket Layer (SSL)
  - Used by client to verify server and protect messages (confid., etc.)

- Initially uses public key encryption to verify server and then switches over to secret session key

- Q: what happens in what order – TCP handshake, SSL handshake, HTTP request/response?

- Q: how is the public key distributed? who verifies who?
  - Too many public keys for client to have the key for each server
Certificates (Public Key Authentication Chains)

- Use a trust hierarchy to distribute public keys with a Public Key Infrastructure (PKI)

- Encoded as certificates (“CA says public key for X is K”)
  - Certificates issued by Certificate Authorities (CAs)
  - Clients need to trust (have public keys for) a small number of CAs
    - Commonly distributed with OS, browser

- Now server can send client its public key in a certificate
  - Client can verify the public key by checking certificate
  - Then client can use public key, e.g., to verify server in SSL
X.509 Certificates

- **X.509**
  - Certificate format standard, global namespace
- Widely used, e.g., in Web browsers
  - Comes with 100 CAs?
Unfortunately: Public Key Revocation

• What if a private key is compromised?
  – Hope it never happens?

• Need certificate revocation list (CRL)
  – And a CRL authority for serving the list
  – Everyone using a certificate is responsible for checking to see if it is on CRL
  – ex: certificate can have two timestamps
    • one long term, when certificate times out
    • one short term, when CRL must be checked
    • CRL is online, CA can be offline
Microsoft Security Bulletin MS01-017
Erroneous VeriSign-Issued Digital Certificates Pose Spoofing Hazard

**Originally posted:** March 22, 2001
**Updated:** June 23, 2003

**Summary**
**Who should read this bulletin:**
All customers using Microsoft® products.

**Impact of vulnerability:**
Attacker could digitally sign code using the name "Microsoft Corporation".

**Recommendation:**
All customers should install the update discussed below.

**Technical description:**
In mid-March 2001, VeriSign, Inc., advised Microsoft that on January 29 and 30, 2001, it issued two VeriSign Class 3 code-signing digital certificates to an individual who fraudulently claimed to be a Microsoft employee. The common name assigned to both certificates is "Microsoft Corporation". The ability to sign executable content using keys that purport to belong to Microsoft would clearly be advantageous to an attacker who wished to convince users to allow the content to run.

The certificates could be used to sign programs, ActiveX controls, Office macros, and other executable content. Of these, signed ActiveX controls and Office macros would pose the greatest risk, because the attack scenarios involving them would be the most straightforward. Both ActiveX controls and Word documents can be delivered via either web pages or HTML mails. ActiveX controls can be automatically invoked via script, and Word documents can be automatically opened via script unless the user has applied the Office Document Open Confirmation Tool.
Update Available to Revoke Fraudulent Microsoft Certificates Issued by VeriSign

View products that this article applies to.

This article was previously published under Q293811

On This Page

SUMMARY

Important Notes

MORE INFORMATION

SUMMARY

In March, 2001, VeriSign, Inc. announced that it had issued two digital certificates to an individual who fraudulently claimed to be a Microsoft employee. This issue is discussed at length in Microsoft Security Bulletin MS01-017. VeriSign has revoked these certificates, and they are listed in the current VeriSign Certificate Revocation List (CRL). However, because the VeriSign code-signing certificates do not specify a CRL Distribution Point (CDP), it is not possible for any browser's CRL-checking mechanism to locate and use the VeriSign CRL. Microsoft has developed an update that rectifies this problem. The update package includes a CRL that contains the two certificates, and an installable revocation handler that consults the CRL on the local computer, rather than attempting to use the CDP mechanism.
Network layer: IPSEC

- IPSEC builds security into IP
  - Protects messages between two computers with confidentiality, integrity/authenticity
  - Commonly used for “VPNs” to set up secure tunnels

- Sets up security associations (SA) between hosts
  - IKE protocol used to set up keys; results in secret session keys

- Adds new headers to IP packets to carry SA and parameters
  - IP has become connection-oriented!
**IPsec**

ESP (Encapsulating Security Payload) provides secrecy and integrity; expands on AH

- Adds ESP header and trailer; inserted after IP header in transport or before in tunnel mode
Link layer: 802.11

- WiFi Protected Access (802.11i/WPA2) is part of 802.11
  - Essential for wireless since transmissions are broadcast!
  - Protection of frames at the link layer
  - Note: old WEP design is broken and easily cracked

- Common design:
  - Clients have password or credentials (one password for home network)
  - Clients authenticate themselves to infrastructure
  - Session keys are set up to encrypt packets
  - One key per client, plus a key for broadcast to all clients
Which layer is best?

- For example, do we want SSL, WPA2, or both?
- Hint: consider what they do/don’t protect
- Typically want end-to-end security and network security
2. User threat: Social engineering

• Con person into giving out information!

• Phone assistant, say:
  – “Hi. I’m your company’s IT administrator. Your boss is currently traveling, and I can’t reach them. I need their password to verify their account hasn’t been broken into. This is really urgent.”

• Somebody phones you, and says:
  – “Hi. I’m with the Bank of America credit card fraud division. We’ve detected suspicious activity on your account, and we want to ensure you haven’t become a victim of identity theft. Before we start, I need to verify your identity. What is your bank account number? SSN?”

• Often far more effective than technical attack
  – Requires everyone to be conscious of security issues
Patricia Dunn: I Am Innocent
PALO ALTO, Calif., Oct. 8, 2006

(CBS) The Hewlett-Packard board of directors was a leaky ship. Secret board deliberations were ending up in the press left and right, and it was decided something had to be done.

That something is arguably the most famous leak investigation since Watergate, and because of it Pattie Dunn, who was chairman of the HP board of directors, now faces criminal charges, and could go to jail.

As correspondent Lesley Stahl reports, the charges stem from the use of something called pretexting, where phone records are retrieved by subterfuge and pretense – where someone calls the phone company and pretends to be someone else in order to obtain the records.

The tactic was apparently used to retrieve the phone records not only of HP board members but of reporters as well. Social security numbers were also obtained, board members and journalists were followed, and there was even discussion of planting spies in newsrooms.

On Thursday, Pattie Dunn was booked on four felony counts in connection with the investigation.
3. Application Vulnerabilities

- Network is the vector, not the fundamental weakness
  - Buffer overflows (unchecked input length)
    - Expecting 100 bytes, send lots more
  - SQL injection attacks
  - Open FTP servers that execute code
  - Many, many more…

- Leads to large numbers of compromised machines
Example: SQL Injection

HI, THIS IS YOUR SON’S SCHOOL. WE’RE HAVING SOME COMPUTER TROUBLE.

OH, DEAR - DID HE BREAK SOMETHING? IN A WAY-

DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?

OH, YES. LITTLE BOBBY TABLES, WE CALL HIM.

WELL, WE’VE LOST THIS YEAR’S STUDENT RECORDS. I HOPE YOU’RE HAPPY.

AND I HOPE YOU’VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

XKCD #327
Police swoop in on New Zealand botmaster

By LIAM TUNG, FOR ZDNET AUSTRALIA
Published: November 30, 2007

New Zealand Police this week cracked down on an alleged botnet ringleader in New Zealand, who the FBI claims had illegal control over 1 million computers.

The sweep is part of the FBI's second phase of "Operation Bot Roast"--the same operation which resulted in four felony charges against 26-year-old Los Angeles security consultant John Schiefer.
4. Administrative boundaries

- Administrative boundaries
  - What should we do to secure the boundaries between networks?
    - e.g., one ISP to another, Internet to customer

- Q: what does IP do for us? A: nothing
Firewalls

• “Middlebox” at boundary
  – Scalable point of defense
  – Break/allow connectivity
  – Useful, but brittle
Firewall evolution

• Originally, a fairly basic “necessity” at edge of network
  – Filter packets based on simple rules to block unused services
  – e.g., if the incoming IP packet contains a TCP SYN to port 80, then allow it only if the IP destination is the Web server machine

• Problem: a bug in your HTTP server (or its configuration) won’t be caught by a basic firewall! Too “application level”

• Solution: smarter firewalls that reconstruct flows (mimic endpoint) and filter based on content (e.g., virus signatures)
  – Called Deep Packet Inspection (DPI)

• Like all middleboxes, firewalls are architecturally problematic
  – e.g., what if end-to-end encryption is used? Can’t eavesdrop!
Reconstructing Flows

- Let’s say you want to search for the text “USER root”. Not enough to just search the data portion of TCP segments!

(Uh oh… we have to reassemble frags and resequence segs)
Fun with Fragments

Imagine an attacker sends:

1. HDR HDR US

2. HDR ER

3. 1,000,000 unrelated fragments

4. HDR HDR ro

5. HDR ot

Think of the entire campus as being a massively parallel computer. That supercomputer is solving the flow-reconstruction problem. Now we’re asking a single host to try to solve that same problem.
ISP boundaries

- Common kinds of functions:
  - Accounting
  - Check IP addresses (ingress filtering, e.g., uRPF)
  - Filter routes (BGP policy)
  - Block “control traffic” with routes and over multiple hops


- These functions are implemented with packet filtering rules.
5. Co-opting/Abusing protocols

• Protocols can often be co-opted or otherwise abused
  – Even when they are implemented correctly; no bugs

• “Don’t think of TCP as a protocol, think of it as an opportunity,”
  – Stefan Savage on Sting tool

• Sometimes this is handy for innovation, e.g., traceroute

• Sometimes this is a security or resource allocation problem
  – E.g, DDOS floods, DNS poisoning
Examples across protocols

- **IP** (packet format, affects forwarding)
  - Can send anything, anywhere, e.g., spoof source address
  - Leads to packet floods, denial-of-service
  - Amplify with broadcast
- **TCP** (allocates bandwidth, server resources)
  - Can send or ACK aggressively; other connections pushed aside
  - Can tie up server state (SYN floods and 3-way handshake)
- **IP/ICMP** (returns error messages)
  - Can trigger unwarranted error messages, concealing source
  - Can tie up host resources (fragments that don’t reassemble)
- **DNS**
  - Can generate fake replies to change host to IP mapping
DNS Attacks

• Cache poisoning:
  – Ask for EVILHOST.COM (say, because of spam)
  – EvilHost.com’s DNS server complies, but also “just happens” to tell you the IP of BankOfAmerica.com
  – DNS client puts it in cache. Fun!

• Spoofing:
  – How does DNS match replies to requests?
  – A 16-bit identifier. So send replies guessing the right identifier!

• DNSSEC
  – A design being deployed that adds security to validate DNS operation
TCP Layer Attacks / SYN flood

- TCP SYN Flooding
  - Exploit state allocated at server after initial SYN packet
  - Send a SYN and don’t reply with ACK
  - Server will wait for 511 seconds for ACK
  - Finite queue size for incomplete connections (1024)
  - Once the queue is full it doesn’t accept requests

- Solution: “SYN Cookies”
  - Construct a special sequence number that has connection info “encrypted”
  - Client sends it back with the ACK; re-encrypt and make sure it matches
  - Makes servers less vulnerable
(Remember the 3-way handshake)
IP Denial of Service

- Attacker can deny service to legitimate users if they can overwhelm the system providing the service
  - System is full of bugs … just send it packets that trigger them
  - System has limited bandwidth, CPU, memory, etc. … just sent it too many packets to handle

- Big issue in practice and lack of effective solutions
  - Today, patch as found (CERT) or build implementation to tolerate DOS
  - Tomorrow, design protocols to withstand, possibly network support for shutting down attack?

- Two broad classes:
  - Nasty packets trigger implementation bugs, e.g., Ping of Death – patch system
  - Packet floods target bandwidth, CPU, memory resources – no solution!
Distributed DOS (DDOS) floods

- Use automated tools to set up a network of zombies
  - Trin00, TFN, mstream, Stacheldraht, …
Complication: Spoofed Addresses

• Why reveal your real address? Instead, “spoof” it.
  – Can implicate others and appear to be many hosts

• Solution?
  – Ingress filtering (ISPs check validity of source addresses) helps, but has poor incentive patterns and is not a complete solution

• Opportunity: “backscatter analysis”
  – Host responds to spoofed packet, sends response packet to essentially random IP
  – If you have a large number of unused IPs, just listen and you’ll hear the backscatter -- can measure DOS attacks!
Complication: amplification

Attacking System

Ping Flood ("Smurf" attack)

Broadcast Enabled Network

Internet
Routing Attacks

• Only want to accept routing updates from neighbors in network
  – BGP often requires TTL = 255
  – May block routing packets across ISP boundaries
  – And restrict by source address

• Nodes in routing systems place great trust in each other
  – Distance Vector Routing
    • Announce “0” distance to all other nodes or blackhole traffic
  – Link State Routing
    • Can claim direct link to any other routers
  – BGP
    • ASes can announce arbitrary prefix aka “hijacking”
More Security Topics

- **Anonymity**
  - Browsing the Web without ISP knowing what you are up to
  - A stronger form of privacy than we usually get

- **Anti-censorship**
  - Preventing the network from filtering out content someone doesn’t like

- **Tracking**
  - Parties building profiles such as your location over time
That’s It – Thank You!