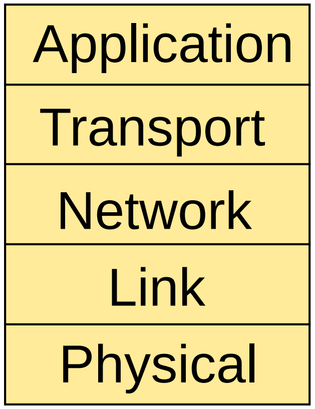
CSEP 561 – Network Security

David Wetherall
djw@cs.washington.edu

Network Security

- Focus
  - How do we secure network systems?

- Topics
  - Message confidentiality/integrity with cryptography
  - Cryptography



Application

Transport

Network

Link

Physical

Overall picture

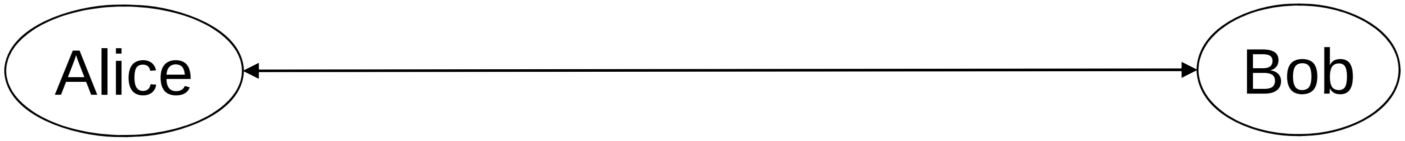
- Security is a generic term, like “performance”
  - Know what you want (security properties)
  - Know what you’re trying to stop (threat / attack model)
- Security is hard
  - It’s a negative goal; can be undone by any weakness (design, implementation, use)
  - Real security is risk management, not mathematics
- The balance today
  - Cryptography is a powerful, principled set of tools at our disposal
  - Exploits come not from breaking the math, but from many, many design flaws (“we used crypto the wrong way”), implementation bugs (buffer overruns), and usage failures (social engineering)
- Take a security course!

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

Attack / Threat Models

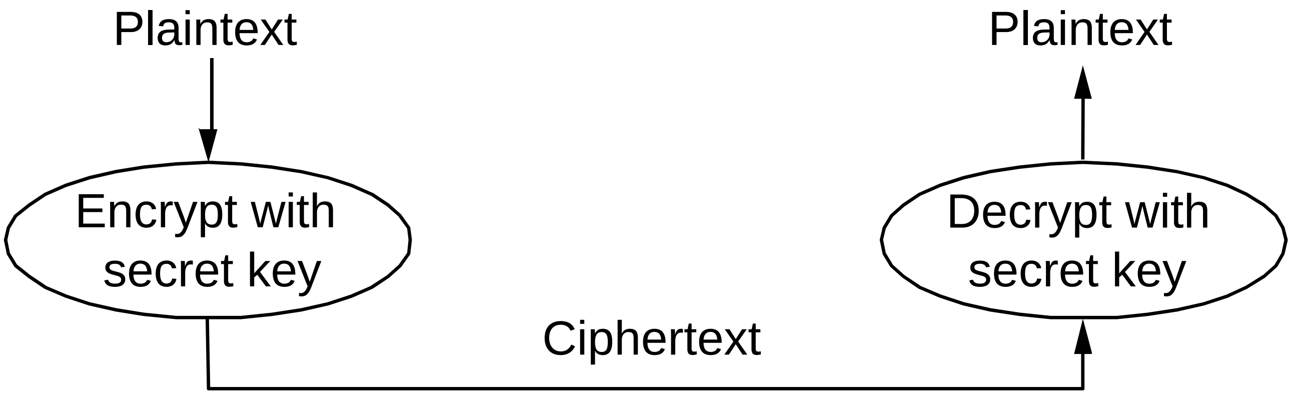


- eavesdropper
- man-in-the-middle
- replay attack
- spoof
- phishing
- …

Privacy / Secrecy

- Main goal: prevent an eavesdropper from understanding what is being sent
- Basic tool is cryptography (encryption). It directly addresses the eavesdropper problem

Secret Key Encryption (AES, 3DES)



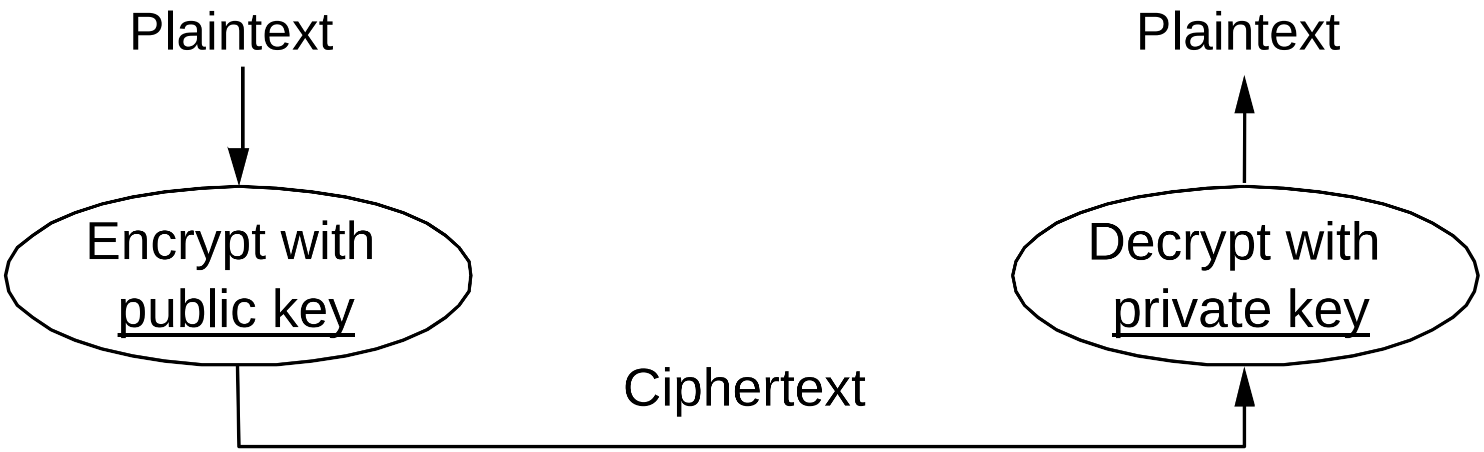
- Also called “shared secret”
- Single key (symmetric) is shared between parties
  - Used both for encryption and decryption
- Pro’s:
  - Fast; hard to break given just ciphertext
- Con’s:
  - key distribution is limiting
    - Suppose you want to create an account at youTube.com?

Encrypting Large Messages

- The basic algorithms encrypt a fixed size block
- Obvious solution is to encrypt a block at a time. This is called Electronic Code Book (ECB)
  - Leaks data: repeated plaintext blocks yield repeated ciphertext blocks
  - Does not guarantee integrity!
- Other modes mix blocks and initialization to avoid this

- An example of what you will learn in a security course

Public Key Encryption (RSA)



- Public key can be published; private is a secret
  - Still have a key distribution problem, though…

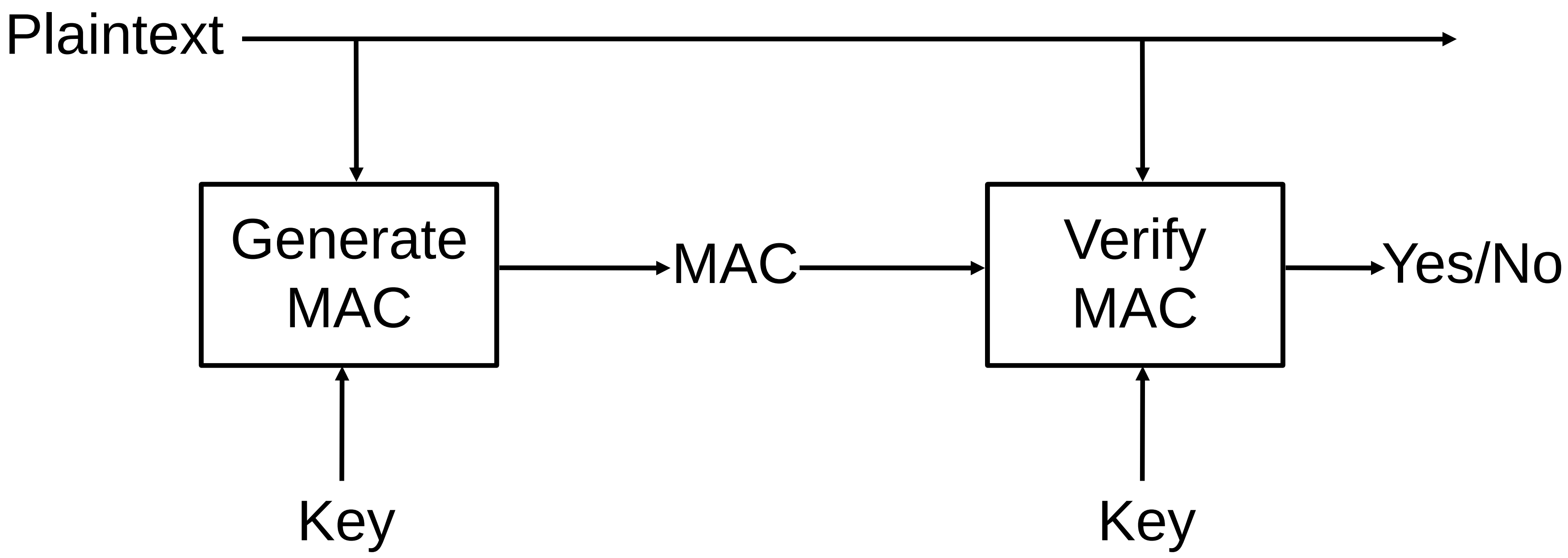
Improving performance

- Public key crypto is slooooow compared to secret key:
  - MD5: 600 Mbps, DES: 100 Mbps, RSA: 0.1 Mbps (from P&D)
- But public key is more convenient & secure in setting up keys
- We can combine them to get the best of both
- Hybrid encryption: encrypt message with random secret key and encrypt secret key with public key.

Integrity & Authenticity

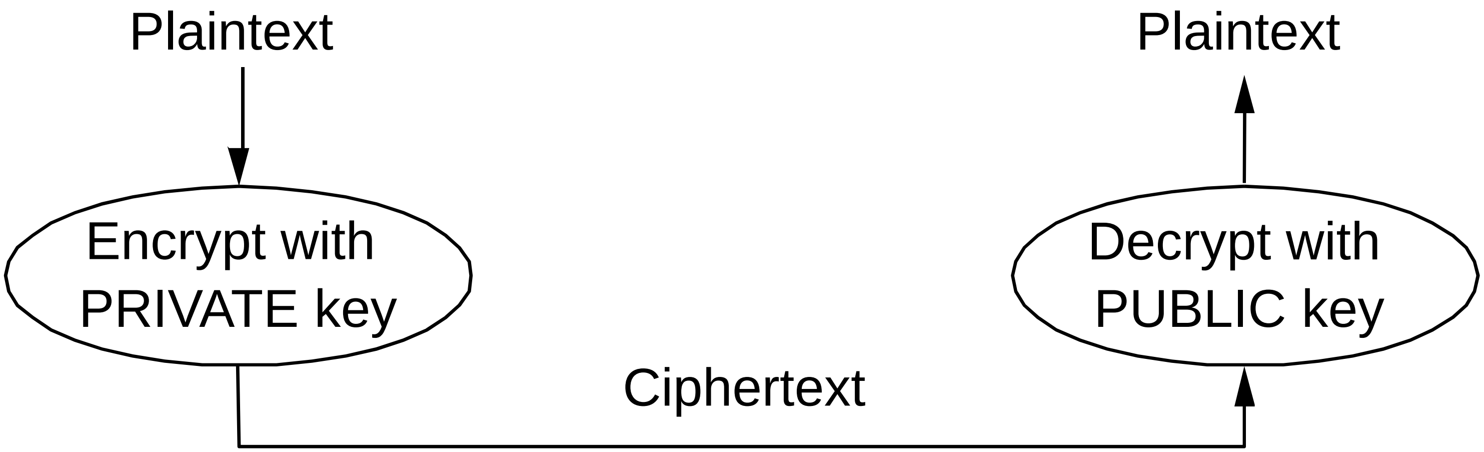
- Main goal: verify that a message has not been altered and that it comes from who it claims
- Message Authentication Code (MAC) allows verifiers (who hold the secret key) to detect changes to content.
  - Sometimes called a MIC, I = Integrity
- Digital signatures allow recipients to verify message integrity and authenticity
- Q: why isn't encryption enough?

Secret Key Integrity



Need to use a different key than for secrecy!

RSA Digital Signature



- Notice that we reversed the role of the keys (and the math just works out) so only one party can send the message but anyone can check it’s authenticity

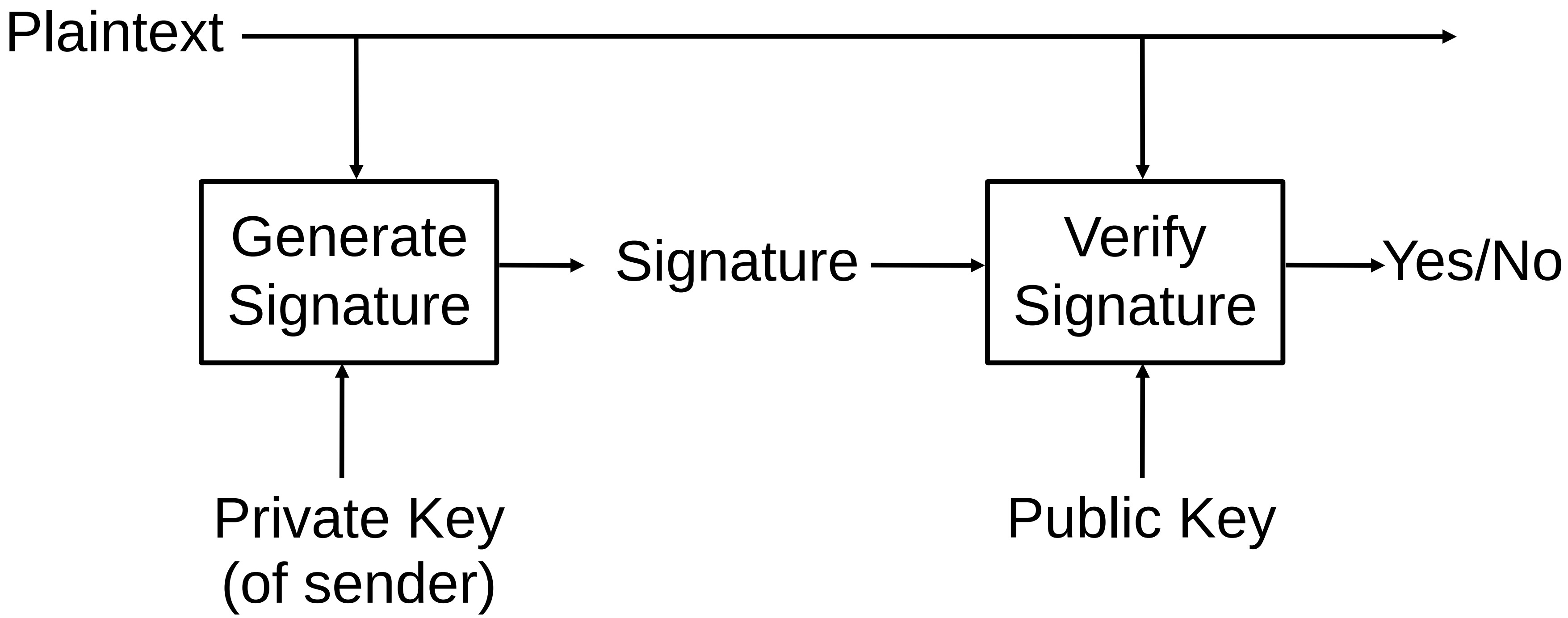
A Faster “RSA Signature”

- Encryption can be expensive, e.g., RSA 1Kbps
- To speed up, let’s sign just the checksum instead!
  - Check that the encrypted bit is a signature of the checksum
- Problem: Easy to alter data without altering checksum
- Answer: Cryptographically strong “checksums”

Cryptographic Hash / Message Digest

- Basically:
  - A hash function (maps arbitrary sized data to a fixed number of bits)
  - Given message M, is cheap to compute
  - Give a hash value, it’s hard to find data that produces that value
    - Ideally, a change to any one bit of the message flips each bit of the hash value with probability 0.5
- Result:
  - Even if the attacker knows the authenticator value, can’t produce bogus data that matches it
- Examples: SHA-1 (160 bits), MD5 (considered broken)

Public Key Integrity Protection



Key Distribution

- These keys need to come from somewhere … Achilles heel
- In a small network, you could get your key from the administrator, just like a password
- But in a large network, we're going to need to trust others to:

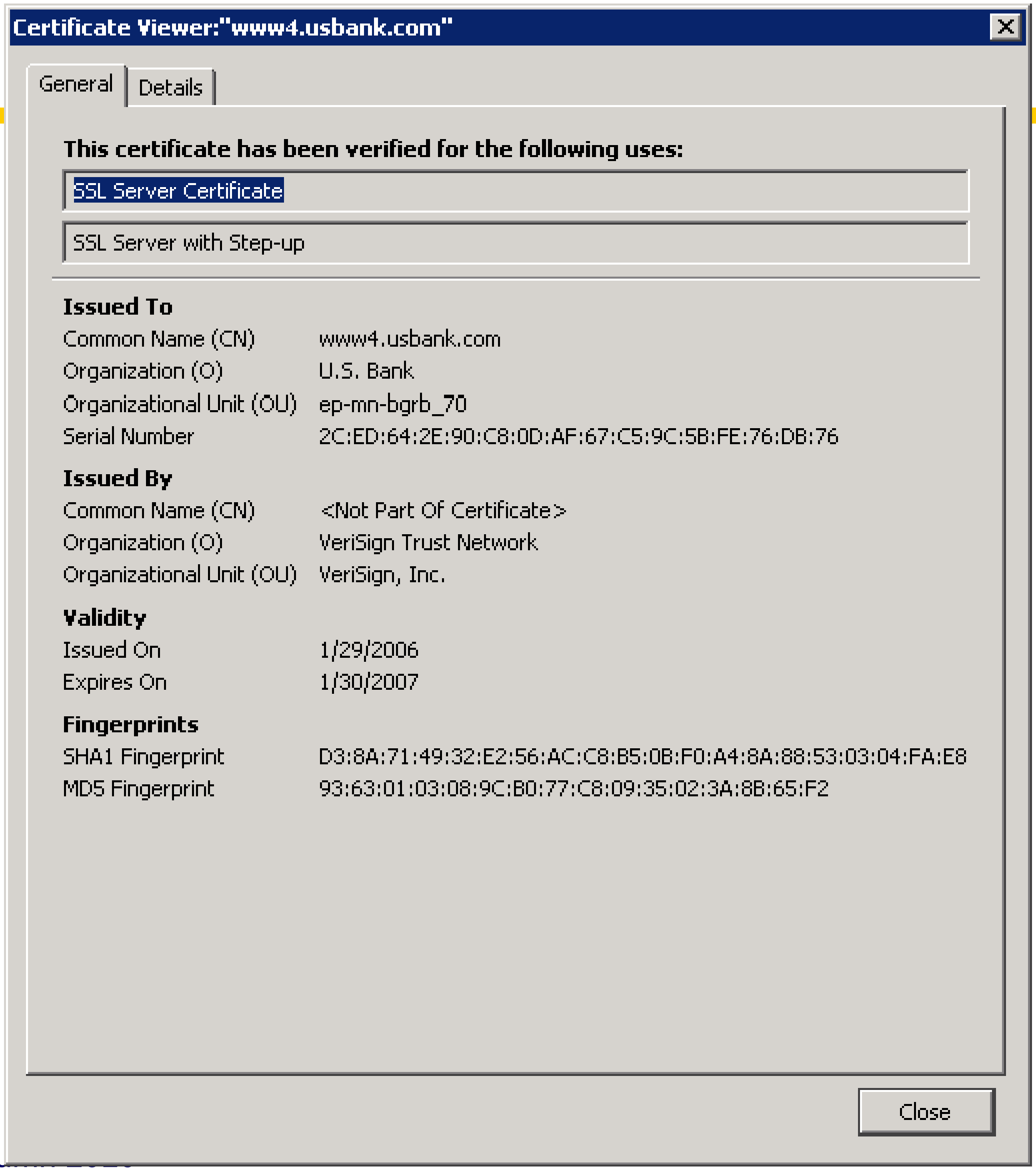
  1) establish shared secrets, or

  2) vouch for public keys.

Public Key Authentication Chains

- Use a trust hierarchy to decide to trust an unknown entity?
- Encoded as certificates ("CA says public key for X is K")
  - Certificates issued by Certificate Authorities (CAs)
  - Clients only need a small number of root CAs
    - Can be distributed with OS, browser
    - Problem is that root CAs have a lot of power!
      - Initial distribution of root CA certificates
- X.509
  - Certificate format standard, global namespace
  - Widely used, e.g., in Web browsers

X.509 Certificates



This certificate has been verified for the following uses:

SSL Server Certificate

SSL Server with Step-up

Issued To

Common Name (CN)	www4.usbank.com
Organization (O)	U.S. Bank
Organizational Unit (OU)	ep-mn-bgrb_70
Serial Number	2C:ED:64:2E:90:C8:0D:AF:67:C5:9C:5B:FE:76:DB:76

Issued By

Common Name (CN)	<Not Part Of Certificate>
Organization (O)	VeriSign Trust Network
Organizational Unit (OU)	VeriSign, Inc.

Validity

Issued On	1/29/2006
Expires On	1/30/2007

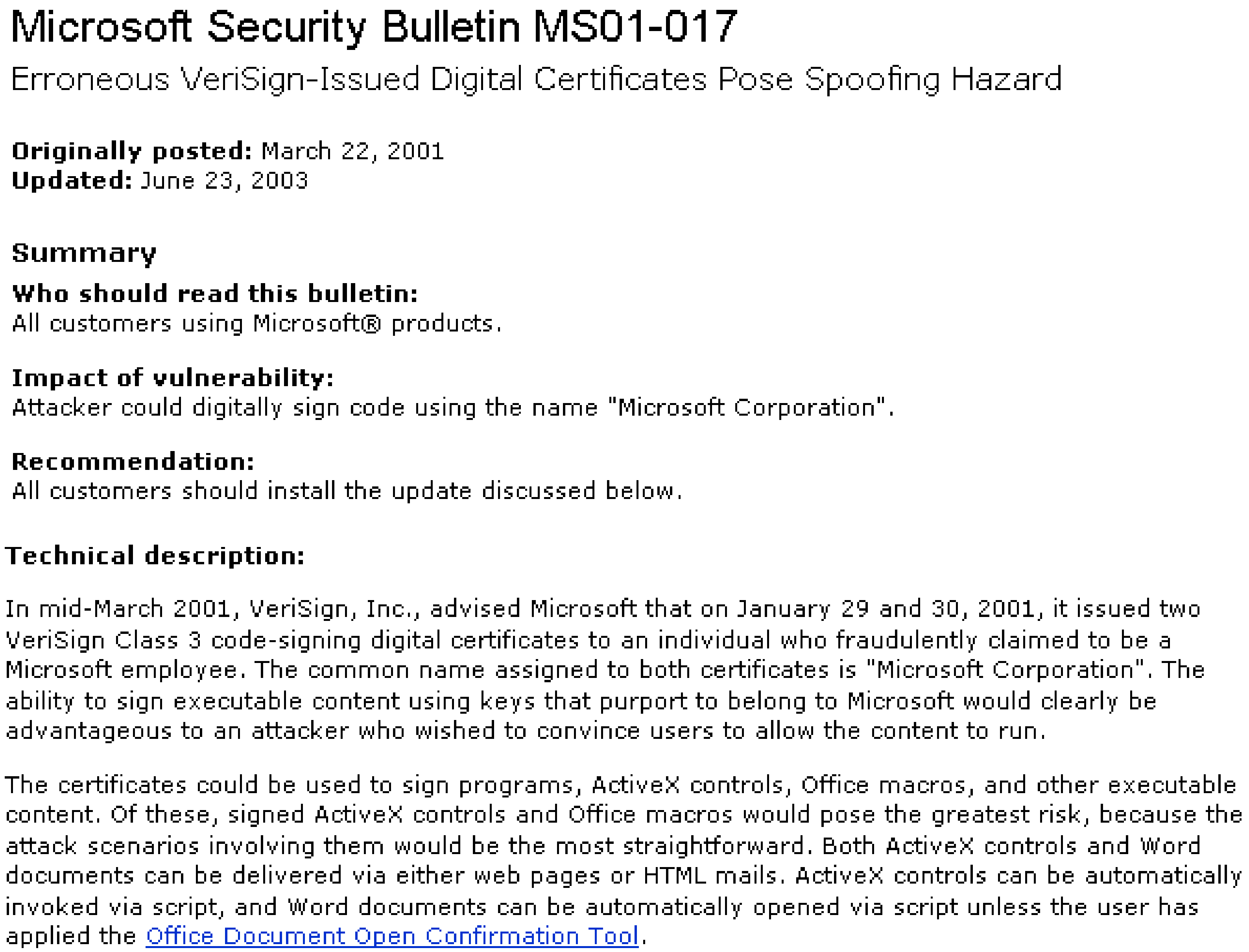
Fingerprints

SHA1 Fingerprint	D3:8A:71:49:32:E2:56:AC:C8:B5:0B:F0:A4:8A:88:53:03:04:FA:E8
MD5 Fingerprint	93:63:01:03:08:9C:B0:77:C8:09:35:02:3A:8B:65:F2

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.

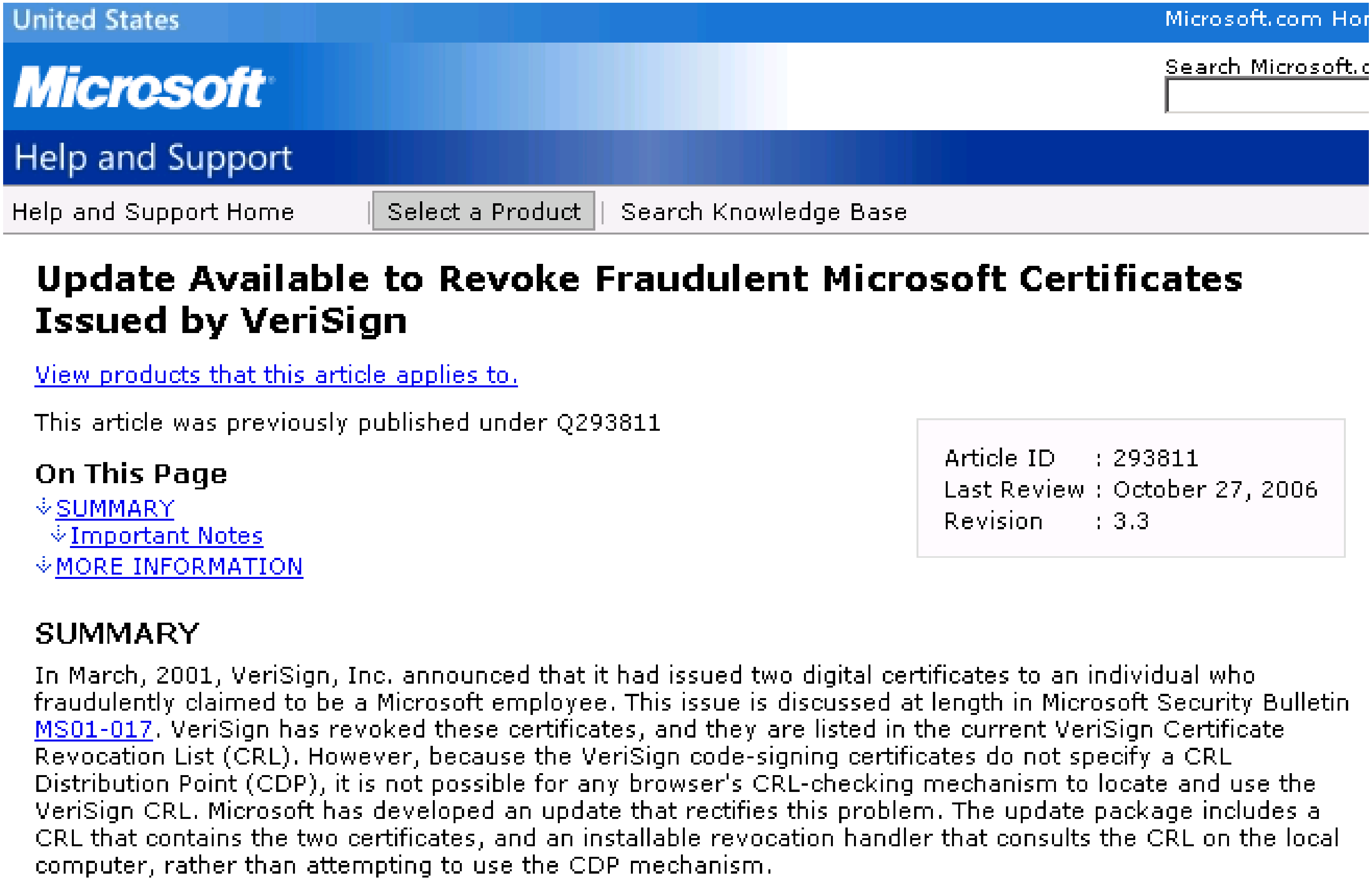
Impact of vulnerability:
Attacker could digitally sign code using the name "Microsoft Corporation".

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

Article ID	:	293811
Last Review	:	October 27, 2006
Revision	:	3.3

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.

Session Keys

- Common to use public keys to authenticate initial contact, then switch to private keys for better performance
  - Secret key is called a session key
  - Ephemeral, lasts only for "the session"

- Example: secure transport layer targeted at Web transactions
  - SSL/TLS inserted between TCP and HTTP to make secure HTTP
  - SSL/TLS uses PKI in the browser and exchanges session keys
  - Client might authenticate Web server but not vice-versa

Network security

1. Security at different layers
   - Link, network, transport, application, human …
2. Application vulnerabilities
   - Buffer overruns, SQL injection …
3. Security at administrative boundaries
   - firewalls, ISPs, VPNs, …
4. Co-opting or abusing network protocols
   - DDOS floods, DNS poisoning, TCP SYN floods, …

Example secure network protocols

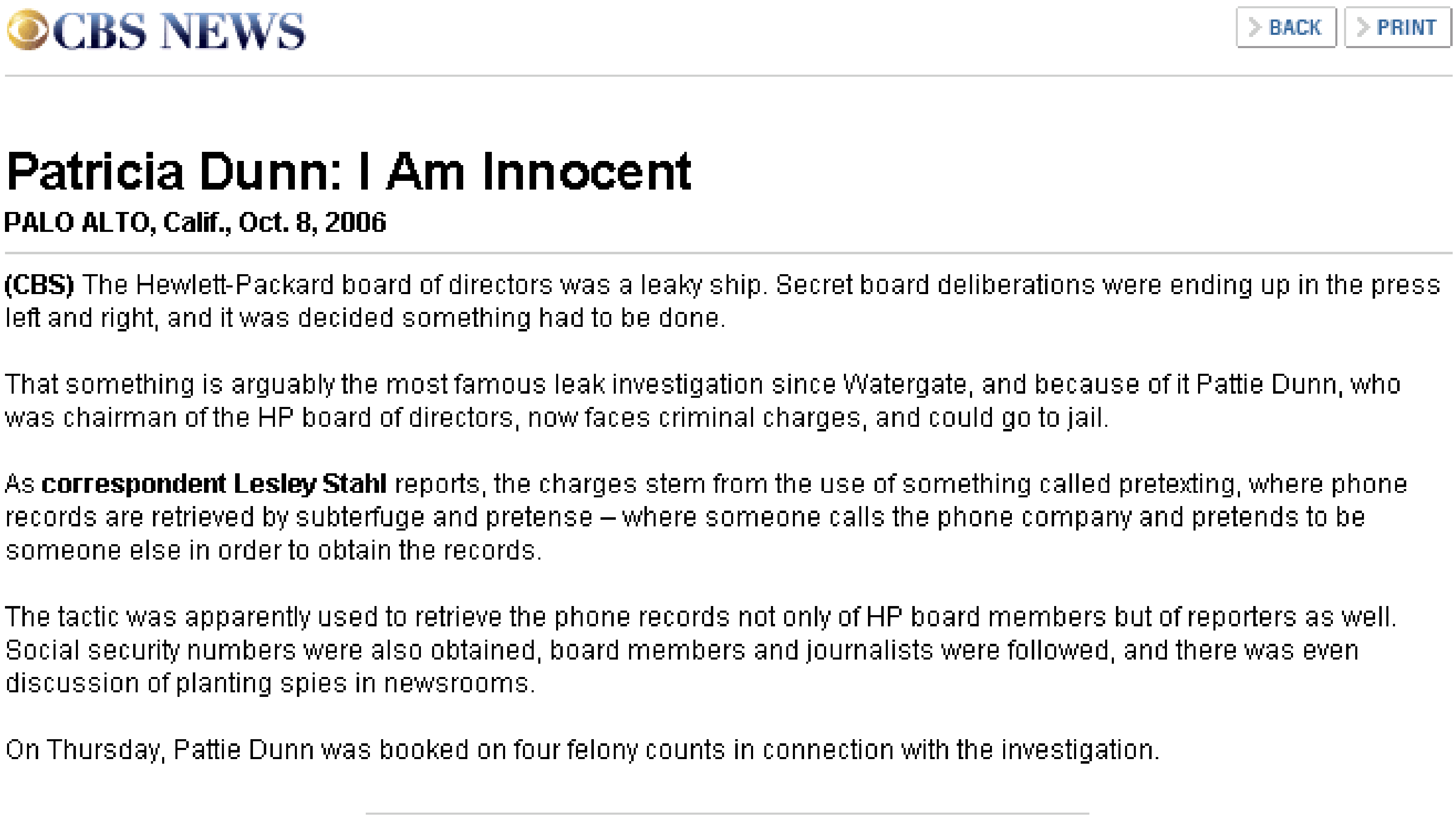
- Cryptography can be applied at multiple layers, top to bottom!
- Secure Shell (ssh)
  - Remote connection with encryption etc.
- Secure Sockets (SSL) and Secure HTTP (HTTPS)
  - For secure Web transactions
- IP Security (IPSEC)
  - Framework for encrypting/authenticating IP packets
- 802.11i / WPA2
  - Protection at the 802.11 link layer

- What layer is “best”?

Highest layer: social engineering

- Con person into giving out information!

- Phone secretary, 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 all people with access to sensitive information 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.

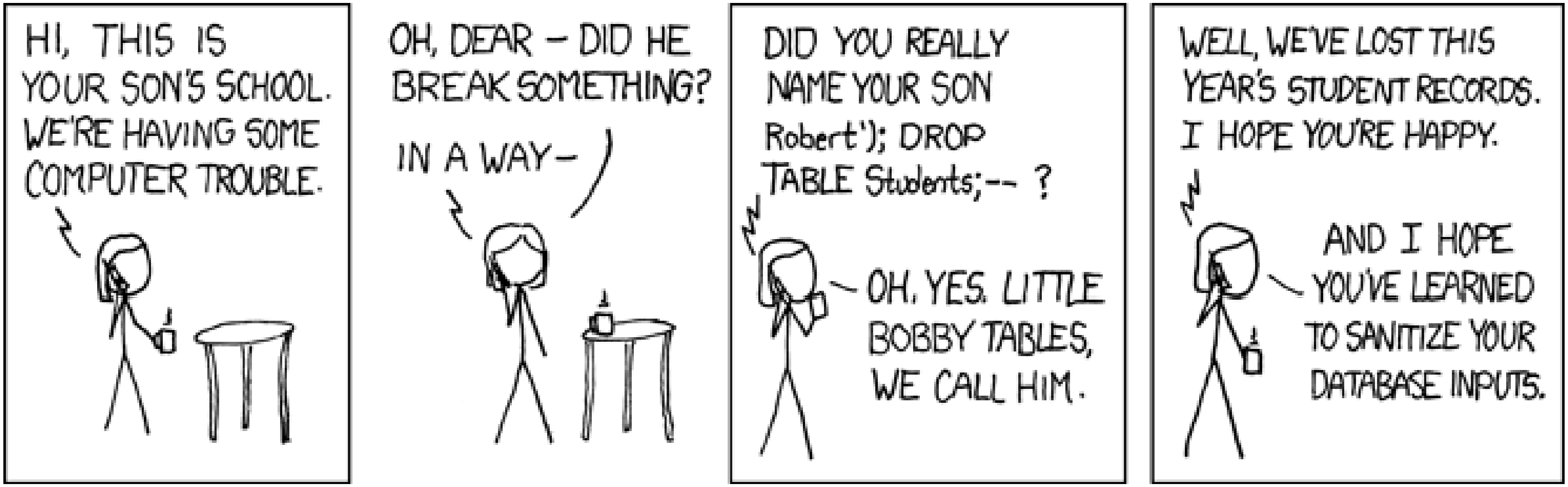
On Thursday, Pattie Dunn was booked on four felony counts in connection with the investigation.

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…

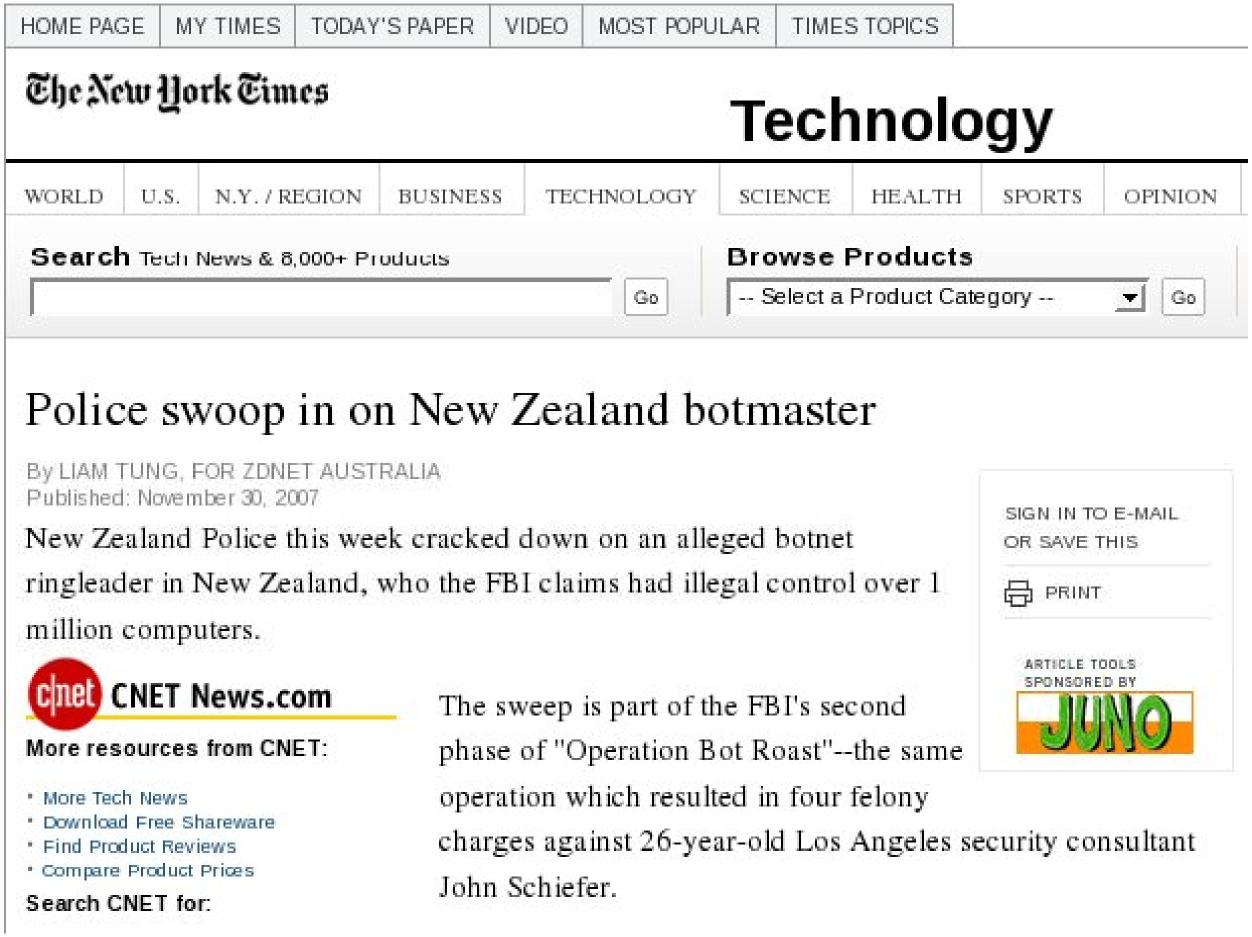
- Leads to large numbers of compromised machines

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

Operation Bot Roast



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.

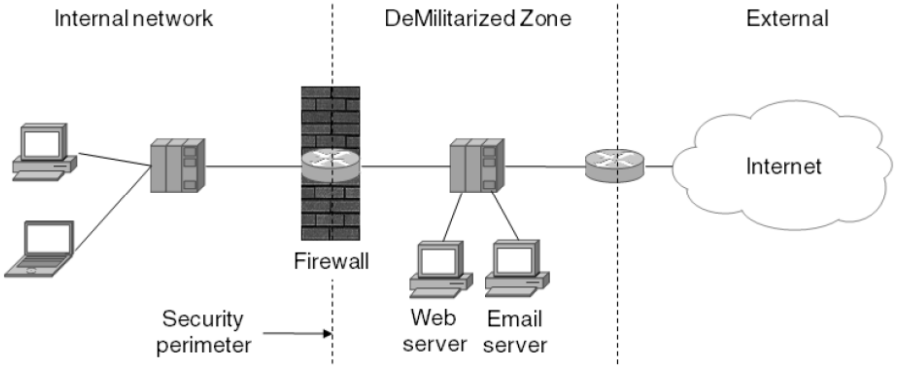
Administrative boundaries

- 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



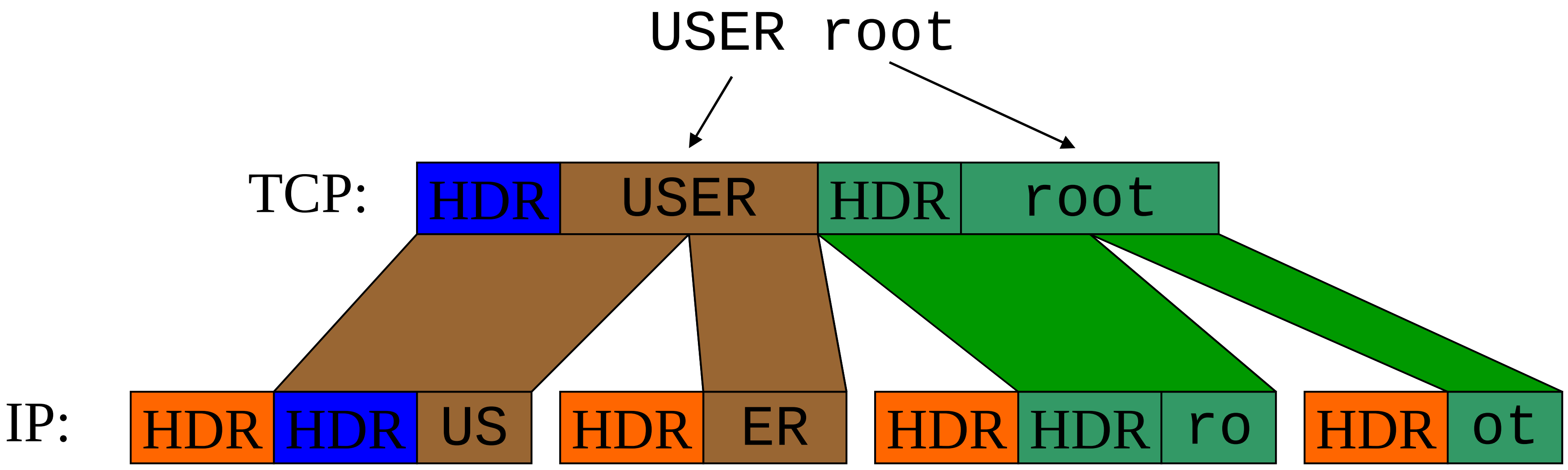
"Oh hey! I just love these things!
...Crunchy on the outside and a chewy center!"
Copyright Gary Larson, 1980. All rights reserved.

Evolution

- Originally, fairly basic: intent was to do per-packet inspection to block unused ports, for example
- Make sure we know exactly what's getting into the network and carefully think about their security
- Problem: a bug in your HTTP server (or its configuration) won't be caught by a basic firewall!
- Later firewalls became smarter – they'd reconstruct the flow. Keep per-flow state (previously impossible)
- Deny, for example, a HTTP request that contains "bobby tables".

Reconstructing Flows

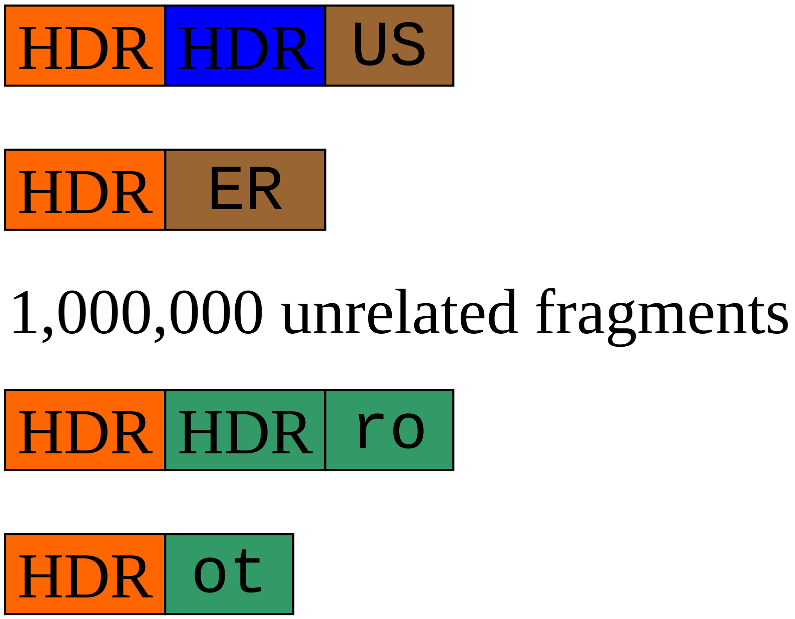
- Let’s say you want to search for the text “USER root”. Is it enough to just search the data portion of TCP segments you see?



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

Fun with Fragments

Imagine an attacker sends:

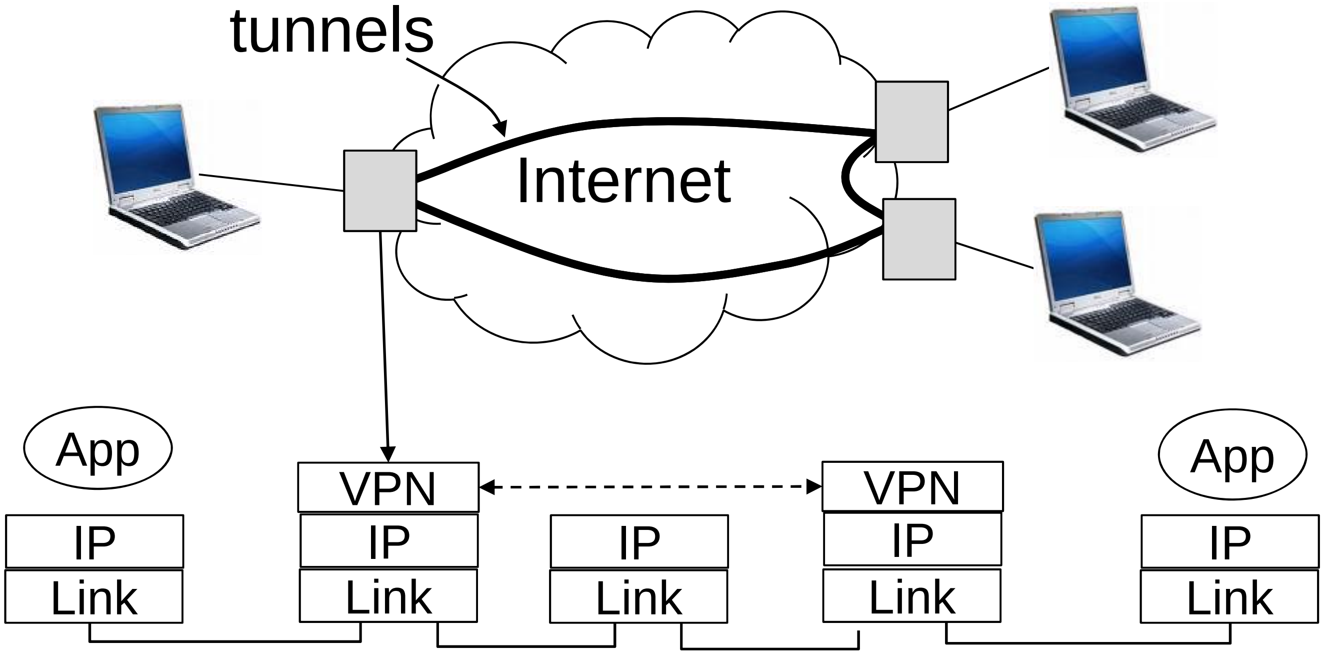


1. HDR HDR US
2. HDR ER
3. 1,000,000 unrelated fragments
4. HDR HDR ro
5. HDR ot

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.

Virtual Private Networks (VPNs)

- Connect a private network via tunnels over the Internet
  - Private network is isolated; tunnels secured, e.g., with IPSEC



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
- Q: What bit of this does IP provide? A: Nothing.

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

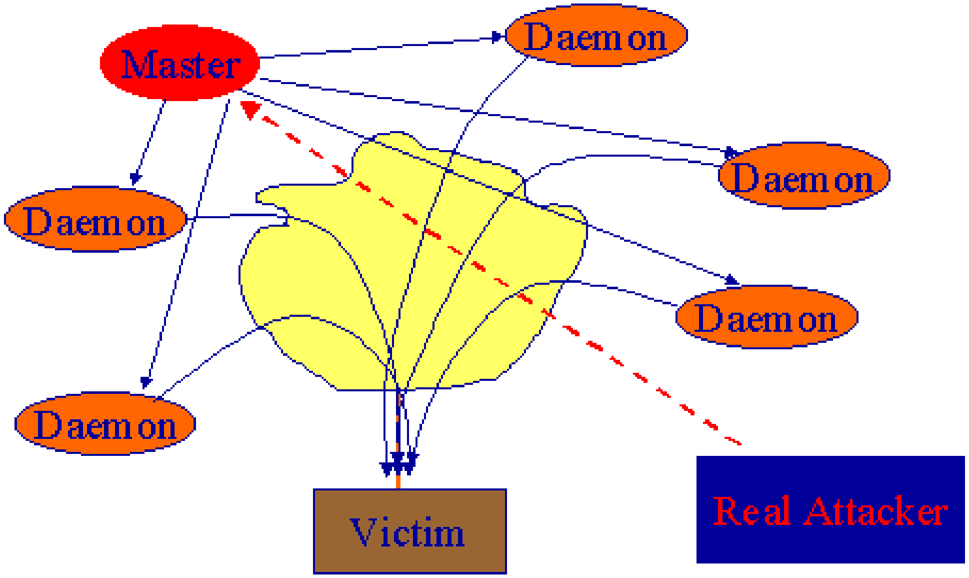
- IP (packet format, affects forwarding)
  - Can send anything, anywhere, e.g., spoof source address
  - Leads to packet floods, denial-of-service
  - Amplify with broadcast
- TCP (allocates bandwidth, server resources)
  - Can send or ACK aggressively; other connections pushed aside
  - Can tie up server 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

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

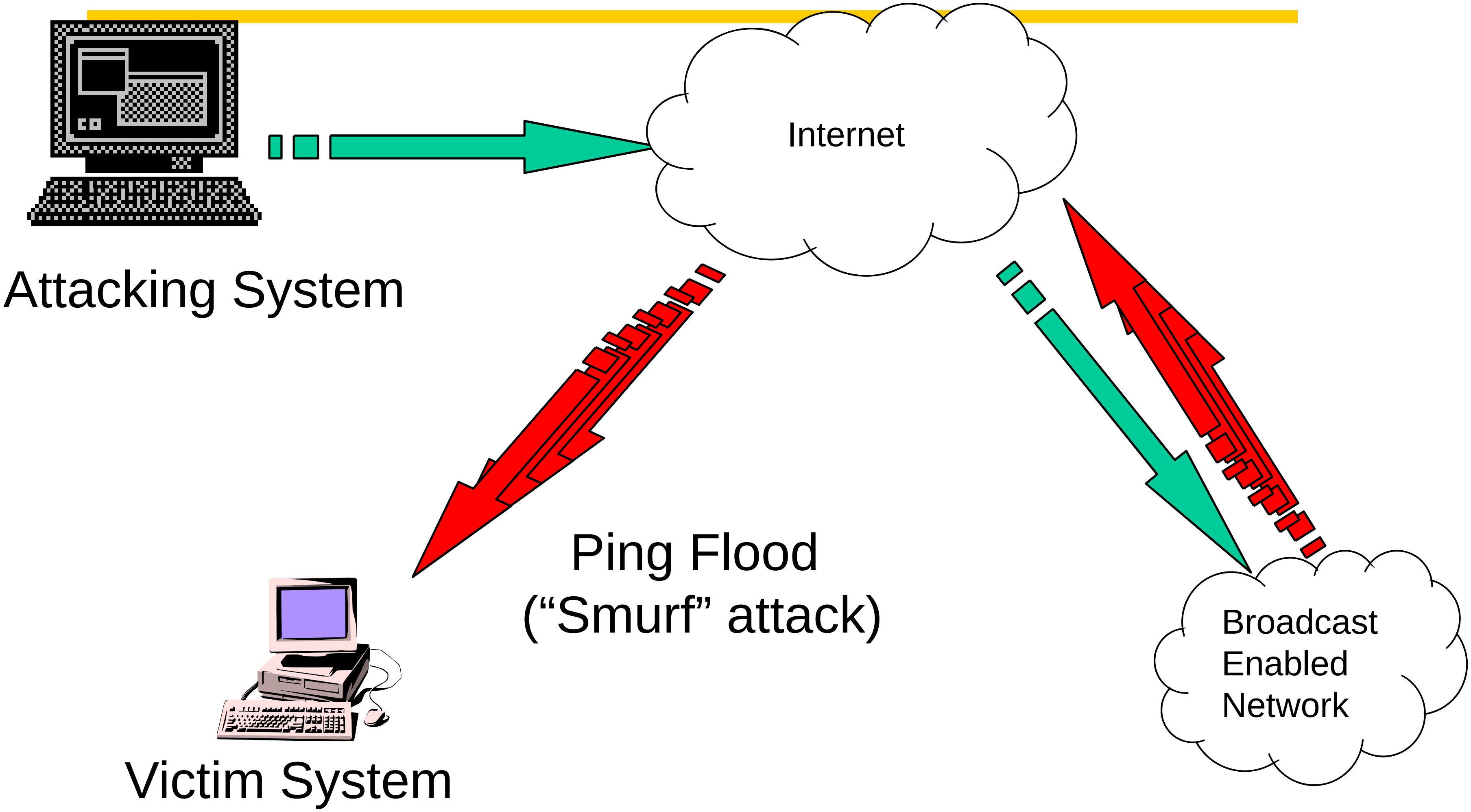
- Use automated tools to set up a network of zombies
  - Trin00, TFN, mstream, Stacheldraht, …



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

Internet

Broadcast Enabled Network

Ping Flood
("Smurf" attack)

Victim System

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

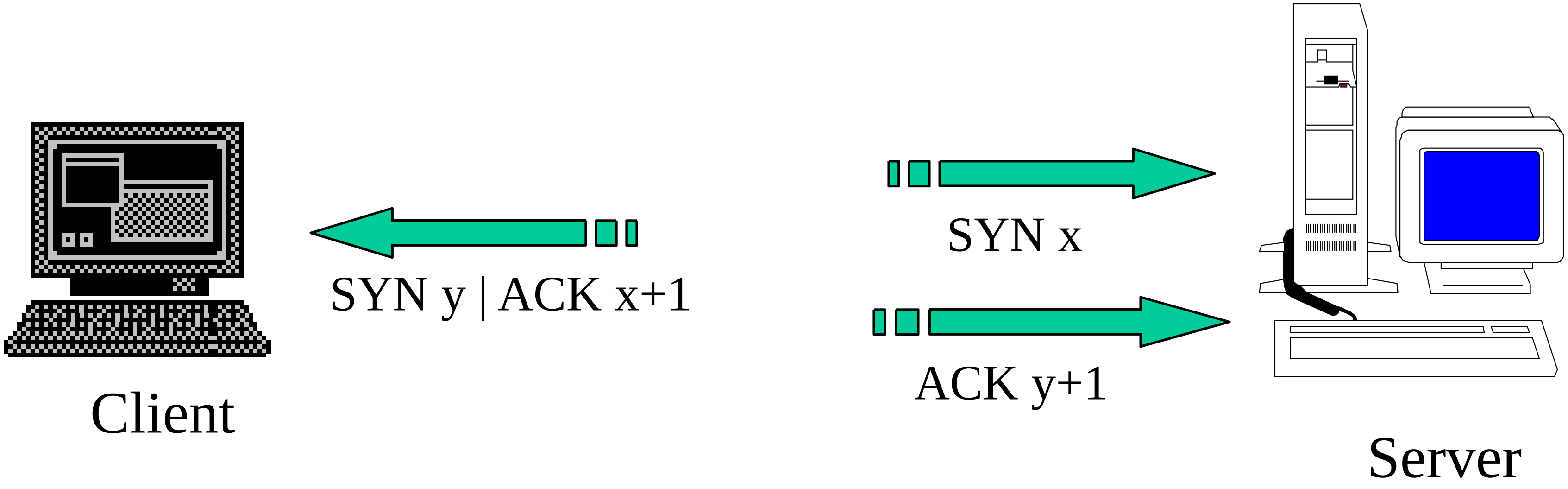
- Nodes in routing systems place great trust in each other
  - Distance Vector Routing
    - Announce “0” distance to all other nodes or blackhole traffic
  - Link State Routing
    - Can claim direct link to any other routers
  - BGP
    - ASes can announce arbitrary prefix aka “hijacking”

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)



SYN x

SYN y | ACK x+1

ACK y+1

Client

Server

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

Misbehaving TCP – significance

- The attacks are significant in theory, but have not been significant in practice
  - Other factors often limit throughput,
    - e.g., Internet access bandwidth, server policies or load

- However, some of these vulnerabilities were important enough to close
  - E.g., modern TCPs may use "byte counting"

Misbehaving TCP – HTTPS

- HTTPS doesn’t help with these attacks. The threat model is different.

- HTTPS:
  - Prevent outsiders from sending/receiving content
  - Authentication/Cryptography is of direct help

- Misbehaving TCP:
  - Prevent insiders from behaving poorly
  - Authentication/cryptography is of no help

Misbehaving TCP – solution costs

- In follow-on papers it turns out that the cost of solutions is very small
  - Minimal bandwidth (1 IP bit and 1 TCP bit/packet) and computation
  - This is somewhat surprising!
- Solutions are now standardized as part of TCP and ECN/IP
  - Deployment is the issue as usual