

CSE 123b Communications Software

Spring 2002

Lecture 16: Network Security

Stefan Savage

Announcements

- Friday office hours rescheduled to 9:30am-10:30am
- Additional office hours next week
 - Monday 1-2pm
 - Wednesday 1-2pm

June 6, 2002

CSE 123b – Lecture 17 – Network Security

2

Overview

- What is network security?
- Communications channel vulnerabilities
 - End-to-end cryptography
- System software vulnerabilities
 - Perimeter defenses
- Protocols vulnerabilities
 - Misinformation
 - Denial-of-service

June 6, 2002

CSE 123b – Lecture 17 – Network Security

3

Network Security?

- What properties do we want?
 - Confidentiality, Integrity, Authenticity
 - Access control
 - Availability
 - Non-repudiation?
 - Consistency?
 - Privacy?
- What is challenging about the network environment?
 - Exposure/sharing
 - Anonymity
 - Fragility

June 6, 2002

CSE 123b – Lecture 17 – Network Security

4

Approaches at 10,000 ft

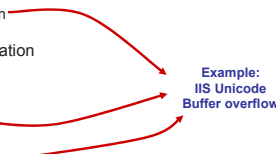
- Physical security
 - Tackle the problem of sharing directly
- “Security through obscurity”
 - Hope no-one will find out what you’re doing!
- Throw math at the problem
 - Cryptography
- Why is security difficult?
 - It’s a negative goal: can you be sure there are no flaws?
 - Often assumptions turn out to be invalid

June 6, 2002

CSE 123b – Lecture 17 – Network Security

5

Taxonomy of attacks

- Vulnerability
 - Design
 - Implementation
 - Configuration
 - Means of exploitation
 - Interception
 - Interruption
 - Modification
 - Fabrication
 - Result
 - Increased access
 - Disclosure of information
 - Corruption of information
 - Denial-of-service
 - Resource theft
- Example:
IIS Unicode
Buffer overflow
- 

June 6, 2002

CSE 123b – Lecture 17 – Network Security

6

Communications channel vulnerabilities

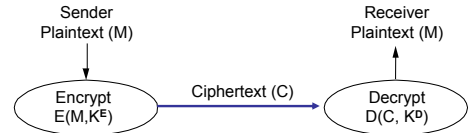
- **Confidentiality**
 - Attacker can intercept messages (passwords, data)
 - Easy on local network; harder at a distance
- **Integrity**
 - Attacker can change messages unbeknownst to sender/receiver
 - Marginally harder attack – must intercept or stop forwarding of legitimate messages
- **Authenticity**
 - Attacker can “pretend” to be a user illegitimately
 - Easy

June 6, 2002

CSE 123b – Lecture 17 – Network Security

7

Basic Encryption for Confidentiality



- Cryptographer chooses functions E, D and keys K^E, K^D
 - Solving $D(C, x) = M$ should be hard without x
- Cryptanalyst try to “break” the system
 - Depends on what is known: E and D, M and C?

June 6, 2002

CSE 123b – Lecture 17 – Network Security

8

Symmetric Key Functions (DES, IDEA)



- $K^E, K^D = K; E(M, K) = \{M\}^K, D(\{M\}^K, K) = M$
- Key must be communicated to both parties, but must be secret to everyone else (key distribution problem)
- Encryption/decryption fast and have equivalent cost
- Also called secret-key or shared-key cryptography

June 6, 2002

CSE 123b – Lecture 17 – Network Security

9

Asymmetric Key Functions (RSA)



- $K^E =$ secret key (SK) $K^D =$ public key (PK)
 - $E(M, SK) = \{M\}^{SK}, D(\{M\}^{SK}, PK) = M$
 - $E(M, PK) = \{M\}^{PK}, D(\{M\}^{PK}, SK) = M$
- DES 100 times faster than RSA in software
 - Typically, PK/SK used to exchange symmetric key, which is used for the conversation

June 6, 2002

PK can be exchanged in the clear (issues?)

10

Integrity (MD5, SHA)

- Verify that a message has not been modified
 - much stronger than checksum (difference?)
- Message digest/ characteristic function/ one-way hash:
 - $H(M) = h$
 - $h, H \neq M$ (inversion resistance) [also called one-way]
 - $M \neq M', \text{ s.t. } H(M) = H(M')$ (collision resistance)
 - Additional mechanism to prevent attacker from also modifying hash
 - » encrypt h, or
 - » $h = H(M, K)$, K is a secret key known by both sender and receiver

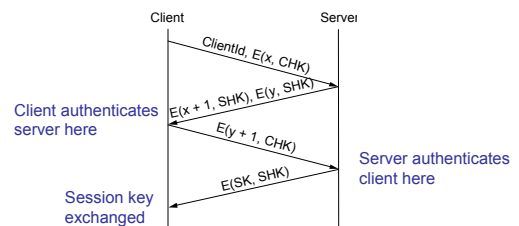
June 6, 2002

CSE 123b – Lecture 17 – Network Security

11

Authenticity Symmetric (secret) keys

- Three-way handshake for mutual authentication
 - Client and server share secrets, e.g., login password



June 6, 2002

CSE 123b – Lecture 17 – Network Security

12

Authenticity Asymmetric (public) keys



- 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

June 6, 2002

CSE 123b – Lecture 17 – Network Security

13

Digital signatures

- Encryption can be expensive, e.g., RSA 1Kbps
- To speed up, let's just encrypt the message digest/hash instead!
- Absolutely critical that hash is "cryptographically strong"
 - Inversion resistance, collision resistance
 - Related to size of hash

June 6, 2002

CSE 123b – Lecture 17 – Network Security

14

Example: SSL

- Transport layer secure channel
- Connection setup
 - Negotiate encryption algorithm
 - Server provides SSL certificate
 - » Certification Authority (CA), principal, principals public key, and timeout
 - Client validates certificate (digital signature) using well-known public-key for CA ,
 - If valid, can use principal's public key to negotiate session key
- Symmetric session key used to encrypt channel
- Who is trying to establish trust with whom here?

June 6, 2002

CSE 123b – Lecture 17 – Network Security

15

System-level vulnerabilities

- How often is security break caused by breaking crypto?
 - Why/where is strength/weakness of crypto important?
- Implementation bugs principal technical source of host compromises
 - Buffer overflow
 - Unchecked parameters
 - Randomness assumptions
 - Race condition
- Ideally: patch/fix all the hosts so no vulnerabilities can be exploited

June 6, 2002

CSE 123b – Lecture 17 – Network Security

16

Perimeter defenses

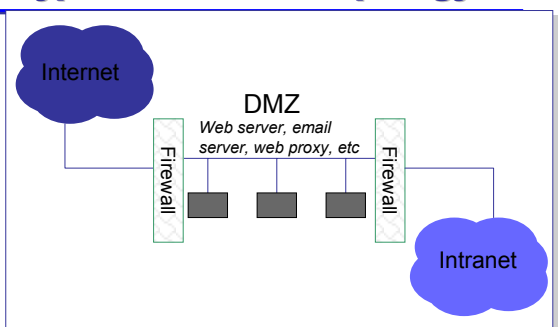
- Key ideas:
 - Too hard to secure/patch/fix each individual system
 - Install "watchdog" system at perimeter of network to protect all hosts inside
 - Model: internal machines are trusted, external machines are untrusted
- Network address translation
 - Multiplex internal address space on small number of public IP addresses; internal hosts can't be addressed directly from the outside
- Firewalls
 - Limit access to end hosts (only those hosts/services that must be made public can be accessed from the outside)
- Intrusion detection systems
 - Detect attempts to break into hosts or exploit system vulnerabilities

June 6, 2002

CSE 123b – Lecture 17 – Network Security

17

Typical Firewall Topology



June 6, 2002

CSE 123b – Lecture 17 – Network Security

18

Types of Firewalls

- **Proxy**
 - End host connects to proxy and asks it to perform actions on its behalf
 - » Policy determines if action is secure or insecure
 - Transport level relays (SOCKS)
 - » Ask proxy to create, accept TCP (or UDP) connection
 - » Cannot secure against insecure application
 - Application level relays (e.g. HTTP, FTP, telnet, etc.)
 - » Ask proxy to perform application action (e.g. HTTP Get, FTP transfer)
 - » Can use application action to determine security
 - Requires applications to be modified to use the proxy
 - Considered to be the most secure since it has most information to make decision

June 6, 2002

CSE 123b – Lecture 17 – Network Security

19

Types of Firewalls

- **Packet filters**
 - Set of filters and associated actions that are used on a packet by packet basis
 - Filters specify fields, masks and values to match against packet contents, input and output interface
 - Actions are typically **forward** or **discard** (yes or no)
 - Such systems have difficulty with things like fragments and a variety of attacks
 - Typically a difficult balance between the access given and the ability to run applications
 - » E.g. FTP often needs inbound connections on arbitrary port numbers – either make it difficult to use FTP or limit its use

June 6, 2002

CSE 123b – Lecture 17 – Network Security

20

Types of Firewalls

- **Stateful packet filters**
 - Allocate state for each flow (i.e. each TCP session)
 - Typically allow richer parsing of each packet (variable length fields, application headers, etc.)
 - Actions can include the addition of new rules and the creation of state to process future packets
 - » Often have to parse application payload to determine “intent” and determine security considerations
 - Rules can be based on packet contents and state created by past packets
 - Provides many of the security benefits of proxies but without having to modify applications

June 6, 2002

CSE 123b – Lecture 17 – Network Security

21

Network Intrusion Detection Systems

- Deployed in similar manner to firewalls
 - Frequently not “in-line” (i.e. if IDS fails, traffic continues)
- Observe all packets and check for intrusion attempts
 - Signature detection (e.g. any HTTP packets with “rm -rf” them)
 - Anomaly detection (e.g. unusual sized requests to port 79)
- Issues
 - False negatives, False positives
 - Evading detection
 - Overhead (can be overwhelmed)
 - Still expensive to respond

June 6, 2002

CSE 123b – Lecture 17 – Network Security

22

Protocol vulnerabilities

- Even if two endpoints have authenticity, integrity, & confidentiality that doesn't mean they will behave
 - Where does trust work as a security mechanism?
- Examples
 - Routing protocols
 - TCP congestion control
 - Denial-of-service

June 6, 2002

CSE 123b – Lecture 17 – Network Security

23

Routing attacks

- Problem: Attacker may advertise bogus routes
 - Claim to originate network/host
 - Intercept packets then re-route to true destination
 - May also cause denial-of-service
- Solutions
 - Policy about which routes you believe (don't accept routes for own network); have well-known neighbors
 - Authentication of routing protocol sessions
 - Open research problem to handle this problem efficiently...

June 6, 2002

CSE 123b – Lecture 17 – Network Security

24

TCP Congestion Control with a Misbehaving Receiver [Savage+99]

- Simple Question
 - Can a TCP client influence how fast a TCP server sends it data?
- Simple Answer: Yes!
- Outline:
 - Why this matters
 - The attacks
 - Some countermeasures

June 6, 2002

CSE 123b – Lecture 17 – Network Security

25

The tragedy of the commons

- Internet bandwidth is a shared resource
 - Stability depends on voluntary end-to-end congestion control
 - If an individual host has both the incentive and ability to *cheat* then the entire system fails
- TCP senders (i.e. content servers)
 - Clearly have ability to cheat (send too fast)
 - Not strong incentive to cheap; own the whole commons
 - » Few senders, each high volume, diverse receivers
- TCP receivers (i.e. Web browsers)
 - Clearly have incentive to *steal* bandwidth
 - Not obvious they have ability

June 6, 2002

CSE 123b – Lecture 17 – Network Security

26

Sources of vulnerability

- ACKs *mean* things that they don't *prove*
 - I was sent in response to a data packet
 - That data packet has been received
 - I have received all the data up to X-1
 - I have (still) not yet received data X
- Sender assumes things that aren't necessarily true
 - At most one ACK generated per data packet
 - Every ACK acknowledges a full-sized packet

June 6, 2002

CSE 123b – Lecture 17 – Network Security

27

Vulnerability 1: Bytes vs. Segments

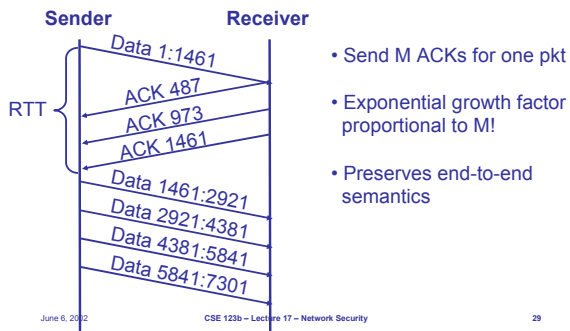
- TCP: reliable byte stream w/ cumulative ACKs
- Cwnd limits unacknowledged data
- TCP begins a session in *slow start*:
During slow start, TCP increments cwnd by at most MSS bytes [one full sized packet] for each ACK received that acknowledges new data.

June 6, 2002

CSE 123b – Lecture 17 – Network Security

28

(1) ACK Division

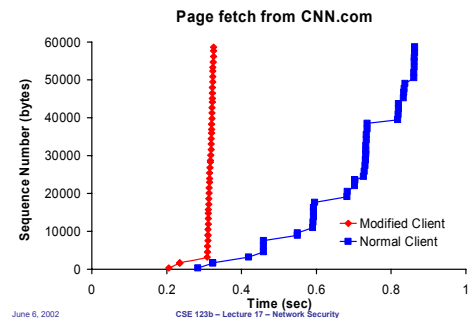


June 6, 2002

CSE 123b – Lecture 17 – Network Security

29

Example



June 6, 2002

CSE 123b – Lecture 17 – Network Security

30

Vulnerability 2: Fast Retransmit and Recovery

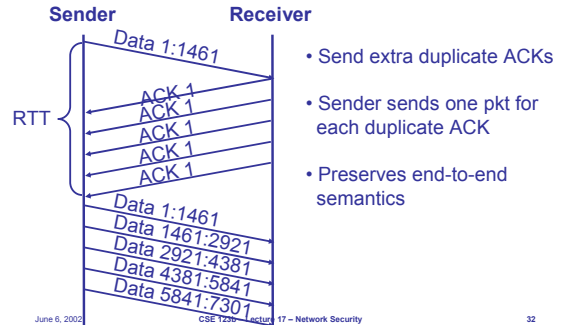
- Receive out-of-order segment => send duplicate ACK
- Sender receives 3 duplicate ACKs => fast retransmits, enters fast recovery
 - $Cwnd = cwnd/2 + 3 * SMSS$
 - On a duplicate ACK, $cwnd += SMSS$
- Each additional duplicate ACK is taken as evidence that a data packet has left the network and therefore $cwnd$ is increased

June 6, 2002

CSE 123b – Lecture 17 – Network Security

31

(2) DupACK Spoofing



June 6, 2002

CSE 123b – Lecture 17 – Network Security

32

Vulnerability 3: Freshness

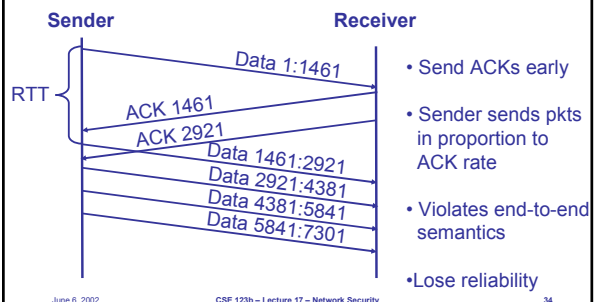
- When sender receives a new ACK, it increases $cwnd$
- *But how do you know the receiver got the data?*
- Must recover at application layer
 - HTTP range request
 - FTP byte request

June 6, 2002

CSE 123b – Lecture 17 – Network Security

33

(3) Optimistic ACKing



June 6, 2002

CSE 123b – Lecture 17 – Network Security

34

Developing a solution

- Not well suited to cryptographic methods
 - Need to ensure **validity** of information, not authenticity or integrity
- Can't enforce behavior at remote peer
- **Solution:** penalize misbehavior
 - Artificially limit connection speed
 - Drop connection

June 6, 2002

CSE 123b – Lecture 17 – Network Security

35

Detecting misbehavior

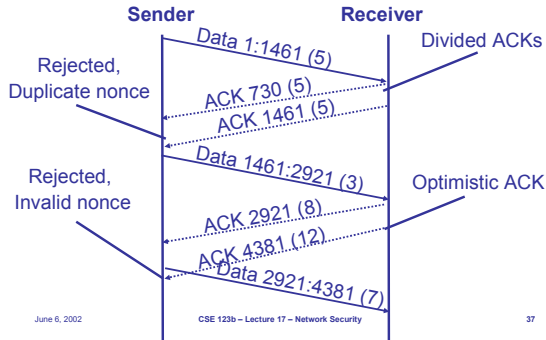
- Eliminate sender assumptions
- Include extra "evidence" in ACK
 - Which data packet it was sent in response to
 - Proof of receipt and proof of freshness
- Mechanism: **cumulative nonce**
 - Sender puts a random # (nonce) in each pkt
 - Receiver echoes sum of nonces
 - Can be implemented probabilistically with a single bit

June 6, 2002

CSE 123b – Lecture 17 – Network Security

36

Cumulative nonce example



How do DoS attacks work?

- Denial-of-service attacks
 - Logic:** exploit bugs to cause crash
 - e.g. Ping-of-Death, Land
 - Flooding:** overwhelm with spurious requests
 - e.g. SYN flood, Smurf
- Distributed denial-of-service attacks**
 - Flooding attack from multiple machines
 - More potent & harder to defend against

June 6, 2002

CSE 123b - Lecture 17 - Network Security

38

Step 1: Attacker infiltrates machines

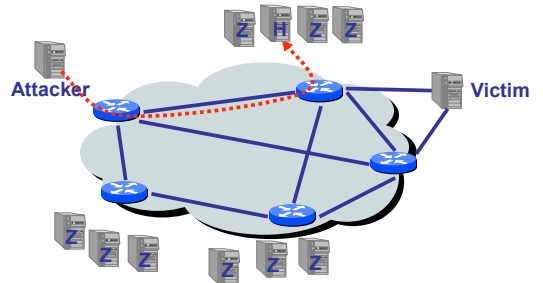
- Scan machines via Internet
- Exploit known bugs & vulnerabilities
- Install backdoor software
 - Zombie software (for attacking target)
 - Handler software (for controlling zombies)
- Cover tracks (e.g. rootkit)
- Repeat... (highly automated)

June 6, 2002

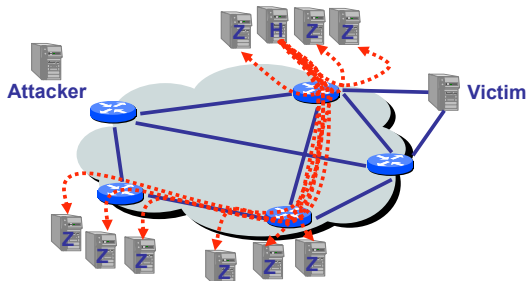
CSE 123b - Lecture 17 - Network Security

39

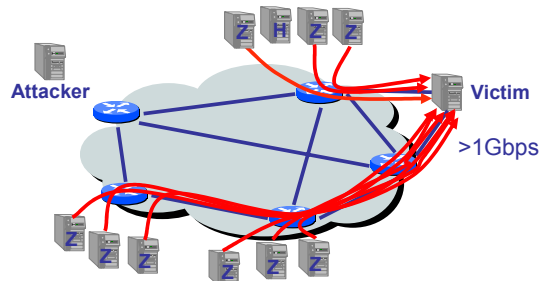
Step 2: Attacker sends commands to handler



Step 3: Handler sends commands to zombies



Step 4: Zombies attack target



Step 5: Victim suffers

- Server CPU/Memory resources
 - Consumes connection state (e.g. SYN flood)
 - Time to evaluate messages (interrupt livelock)
 - » Some messages take "slow path" (e.g. invalid ACK)
 - Can cause new connections to be dropped and existing connections to time-out
- Network resources
 - Routers PPS limited, FIFO queuing
 - If attack is greater than forwarding capacity, good data will be dropped

June 6, 2002

CSE 123b – Lecture 17 – Network Security

43

Simple question

How prevalent are
denial-of-service attacks?

June 6, 2002

CSE 123b – Lecture 17 – Network Security

44

Existing data is anecdotal

Press reports:



Analysts: "Losses ... could total more than \$1.2 billion"
- Yankee Group report

Surveys: "38% of security professionals surveyed
reported denial of service activity in 2000"
CSI/FBI survey

June 6, 2002

CSE 123b – Lecture 17 – Network Security

45

Quantitative data?

- Isn't available (i.e. no one knows)
- Inherently **hard to acquire**
 - Few content or service providers collect such data
 - If they do, its usually considered sensitive
- **Infeasible to collect** at Internet scale
 - How to monitor enough to the Internet to obtain a representative sample?

June 6, 2002

CSE 123b – Lecture 17 – Network Security

46

A good estimate: [Moore, Voelker, Savage01]

- Backscatter analysis
 - New technique for estimating **global** denial-of-service activity
- First data describing Internet-wide DoS activity
 - ~4,000 attacks per week (> 12,000 over 3 weeks)
 - Instantaneous loads above 600k pps
 - Characterization of attacks and victims

June 6, 2002

CSE 123b – Lecture 17 – Network Security

47

Key idea

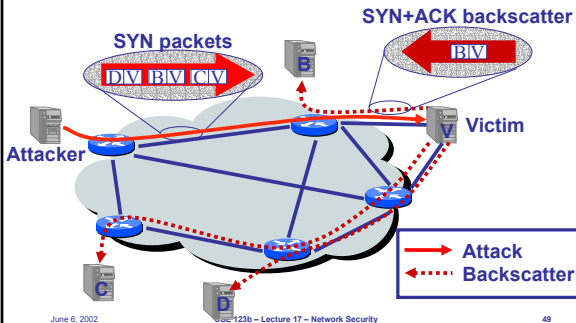
- Flooding-style DoS attacks
 - e.g. SYN flood, ICMP flood
- Attackers spoof source address **randomly**
 - True of all major attack tools
- Victims, in turn, respond to attack packets
- Unsolicited responses (*backscatter*) equally distributed across IP address space
- Received backscatter is **evidence** of an attacker elsewhere

June 6, 2002

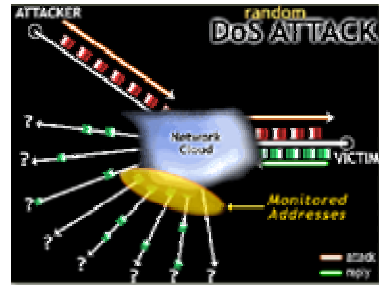
CSE 123b – Lecture 17 – Network Security

48

Random IP spoofing produces random backscatter



Example



Backscatter analysis

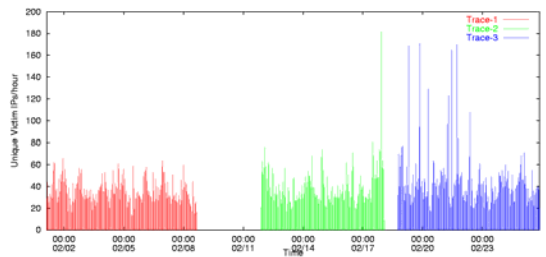
- Monitor block of n IP addresses
- Expected # of backscatter packets given an attack of m packets:

$$E(X) = \frac{nm}{2^{32}}$$

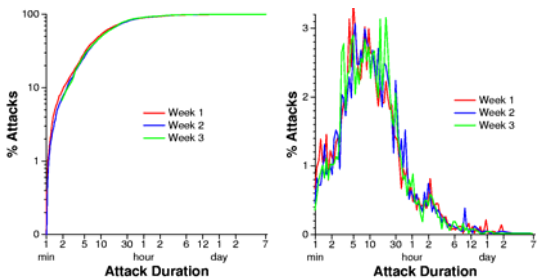
- Extrapolated attack rate R' is a function of measured backscatter rate R :

$$R \geq R' \frac{2^{32}}{n}$$

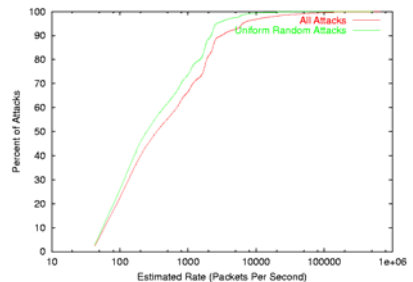
Attacks over time



Attack duration distribution



Attack rate distribution



Victim characterization by DNS name

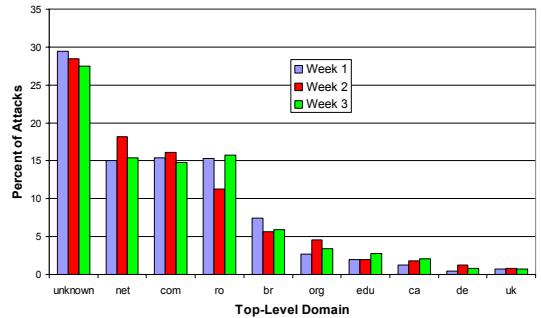
- Entire spectrum of commercial businesses
 - Yahoo, CNN, Amazon, etc and many smaller biz
- Evidence that minor DoS attacks used for personal vendettas
 - 10-20% of attacks to home machines
 - A few very large attacks against broadband
 - Many reverse mappings clearly compromised (e.g. is.on.the.net.illegal.ly and the.feds.cant.secure.their.shellz.ca)
- 5% of attack target infrastructure
 - Routers (e.g. core2-core1-oc48.paol.above.net)
 - Name servers (e.g. ns4.reliablehosting.com)

June 6, 2002

CSE 123b – Lecture 17 – Network Security

55

Victim breakdown by TLD



Denial-of-Service summary

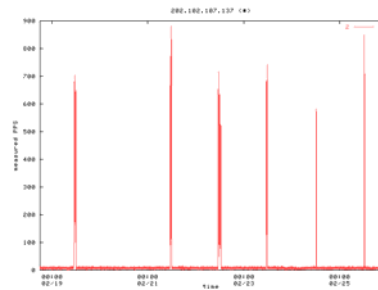
- Lots of attacks – some very large
 - >12,000 attacks against >5,000 targets in a week
 - Most < 1,000 pps, but some over 600,000 pps
- Everyone is a potential target
 - Targets not dominated by any TLD, 2LD or AS
 - » Targets include large e-commerce sites, mid-sized business, ISPs, government, universities and end-users
 - Something weird is happening in Romania
- New attack “styles”
 - Punctuated/periodic attacks
 - Attacks against infrastructure targets & broadband

June 6, 2002

CSE 123b – Lecture 17 – Network Security

57

Example 1: Periodic attack (1hr per 24hrs)

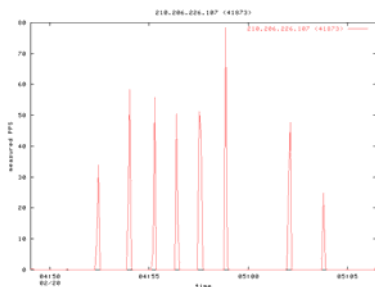


June 6, 2002

CSE 123b – Lecture 17 – Network Security

58

Example 2: Punctuated attack (1min interval)



June 6, 2002

CSE 123b – Lecture 17 – Network Security

59

Other security issues...

- Detecting/tracking denial-of-service attacks
- Statically/dynamically detecting likely program vulnerabilities (e.g. buffer overflows, race conditions)
- Steganography
- Digital watermarking, copy protection
- Revocable credentials
- Secure and/or anonymous storage
- Micropayments
- Side-channel attacks (timing, power, etc)
- Tamper resistant environments
- Worms, viruses, etc...

June 6, 2002

CSE 123b – Lecture 17 – Network Security

60

Summary

- Security is a huge field, poorly fleshed out
- Mostly based on trust
 - Authenticity, confidentiality, integrity to establish trust with outsider
 - Firewalls/IDS define trusted vs untrusted infrastructure
 - If you don't have trust, these measures don't help
- **Every** protocol in use today likely has security holes
 - We don't design for the adversary
- How many of the flaws we discussed today still exist?

Final

- Similar flavor to the midterm
- Covers everything from first day of class until today
- You can have one 8.5x11 sheet (two sides) of notes
 - However, they're unlikely to be super-helpful, the exam is about understanding no regurgitation.
- I haven't written the exam yet, so I can't answer specific questions about its contents...