Network Security

David Parter

University of Wisconsin

Computer Sciences Department

Computer Systems Lab

CS640 27 November 2007

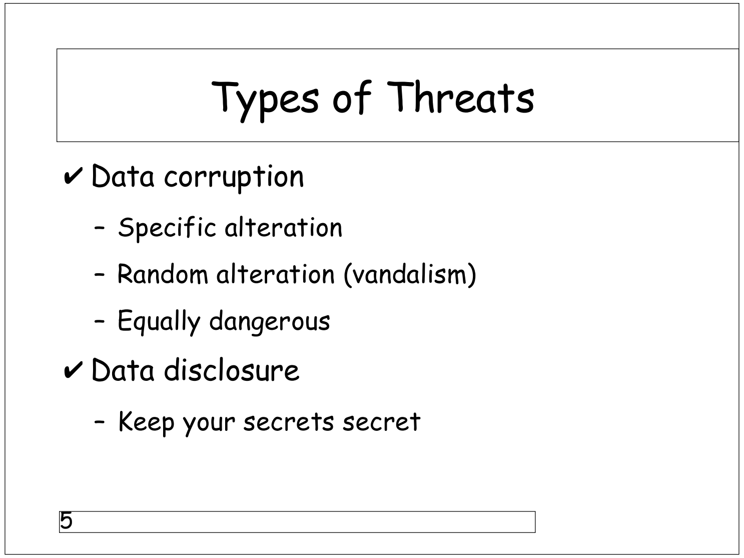
Topics

- ✔ Background: Threats and Security Policies
- ✔ Tools and Defenses:
  - Firewalls
  - Virtual Private Networks
  - Network Intrusion Detection Systems
  - Port Scanning
  - Network & Configuration Management
- ✔ CSL Network Security

Threats and Security Policies

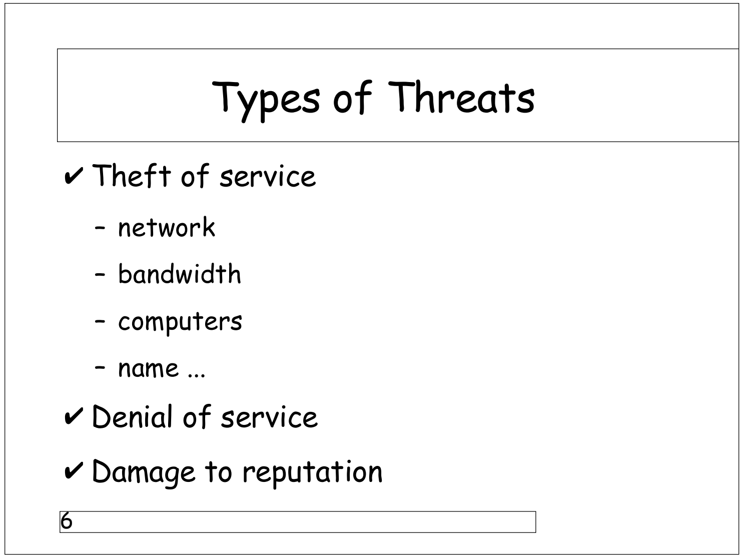
Analyze The Threats

- ✔ Analyze potential threats before choosing a defense
- ✔ Without knowing threats, it is impossible to assess the defenses



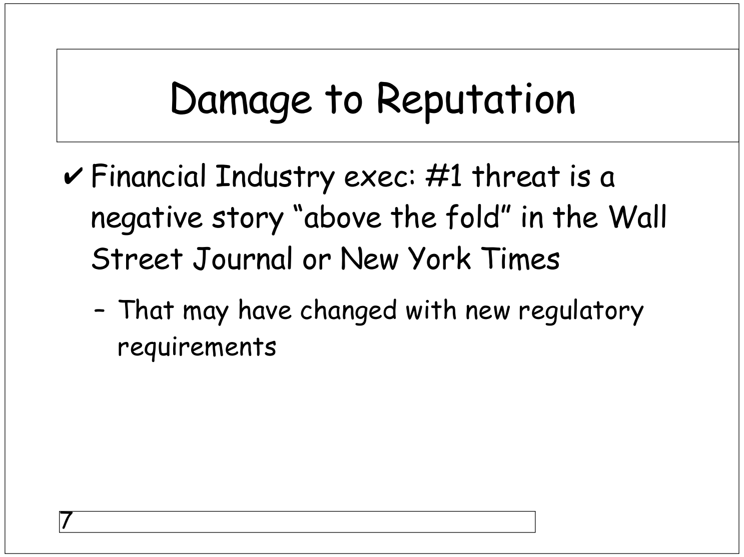
Types of Threats

- ✔ Data corruption
  - Specific alteration
  - Random alteration (vandalism)
  - Equally dangerous
- ✔ Data disclosure
  - Keep your secrets secret



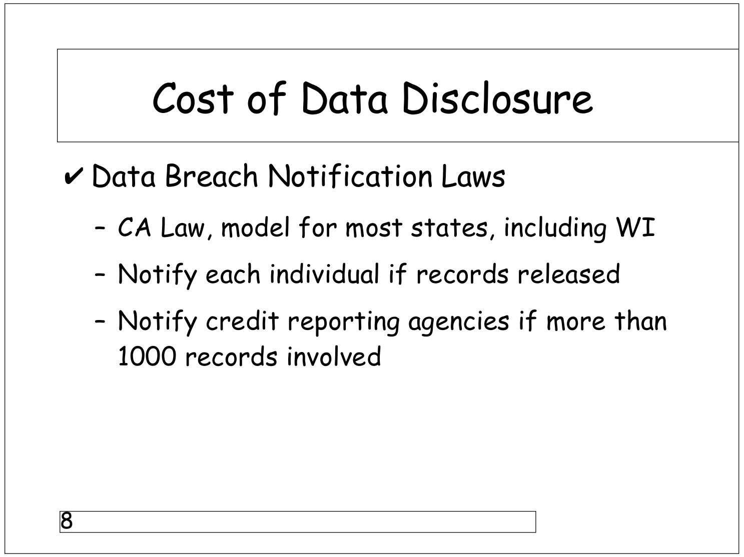
Types of Threats

- ✔ Theft of service
  - network
  - bandwidth
  - computers
  - name ...
- ✔ Denial of service
- ✔ Damage to reputation



Damage to Reputation

- ✔ Financial Industry exec: #1 threat is a negative story “above the fold” in the Wall Street Journal or New York Times
  - That may have changed with new regulatory requirements



Cost of Data Disclosure

- ✔ Data Breach Notification Laws
  - CA Law, model for most states, including WI
  - Notify each individual if records released
  - Notify credit reporting agencies if more than 1000 records involved

Cost of Data Disclosure

- Very likely to be widely reported in the news media
  - Damage to reputation
- Liability/remediation
  - credit monitoring for all individuals?
  - Civil actions?

Example: Medical Industry

- Data corruption & Denial of service:
  - Could lead to incorrect diagnosis, treatment
  - Potentially life-threatening
- Data disclosure
  - Loss of patient record privacy
  - Many potential social, legal and business costs
- Damage to reputation

Example: Financial Industry

- Data corruption
  - Potential for incorrect (or less profitable) stock market trades
  - Account records can probably be reconstructed
- Data disclosure
  - Loss of competitive advantage
  - Violation of securities laws

Example: A University Academic Department

- Data corruption:
  - Loss of experiments/experimental data
  - Incorrect experimental results
- Data disclosure
  - Disclosure of confidential data: human subjects data, industrial partner data, current research, student grades, exams, peer reviews, ...

Security Policies

- After threat analysis, develop security policies
- Policies provide guidance
  - to employees in ongoing operations,
  - to security/system administration staff
- Develop policies before a crisis hits

Tools and Defenses

Firewalls

- Background & Security model
- Type of firewalls
- Firewall rules

References and Resources

- **Firewalls and Internet Security: Repelling the Wily Hacker (2nd ed)**
  Cheswick, Bellovin and Rubin
- **Building Internet Firewalls (2nd ed)**
  Zwicky, Chapman and Cooper
- Firewall-wizards mailing list
  - http://honor.trusecure.com/mailman/listinfo/firewall-wizards

Security Model

- ✔ Perimeter security
  - Like a guard at the gate, checking ID badges
  - Assumes that “inside” is trusted, “outside” is not
  - Larger area “inside” perimeter -> more complexity, weaker security
  - Smaller perimeter -> more specific security
- ✔ Applies predefined access rules

Why Use a Firewall?

- ✔ Protect vulnerable services
  - Poorly designed protocols
  - Poorly implemented protocols/services
- ✔ Protect vulnerable computers/devices
  - Poorly configured
  - Can't be configured
  - Can't be patched

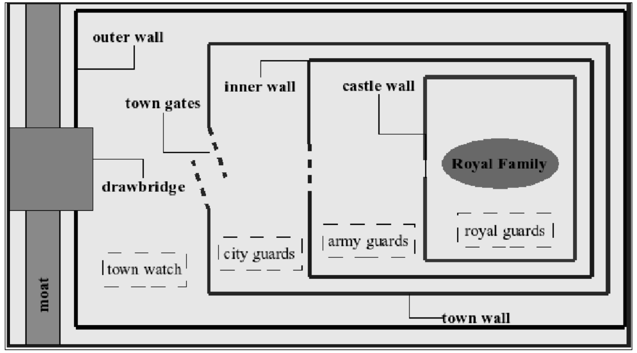
Why Use a Firewall?

- ✔ To protect an “appliance”
- ✔ Protect a system that can not be upgraded
  - Version/upgrade restrictions from vendor
  - ex: printers; data acquisition devices; scientific “instruments”; devices with customized & embedded versions of popular operating systems; devices with embedded web servers for configuration/control ...

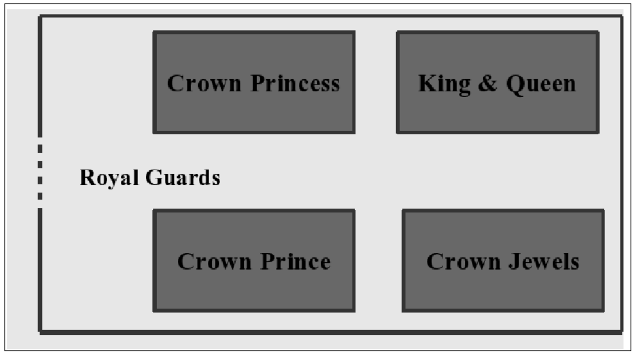
Why Use a Firewall?

- ✔ Defeat some denial of service (DOS) attacks
  - If the firewall has enough bandwidth
- ✔ Considered an “easy” solution
  - Satisfy “check-box” requirements
  - Only need to deal with security in one place (not really an advantage from a total security point of view)

Perimeter Security and Defense in Depth



Improved Security: Reduced Perimeters



Types of Firewalls: Basic Technology options

- ✔ Basic Technology Options:
  - Packet Filtering (screening)
  - Application Proxy
- ✔ Other Factors:
  - Statefull vs. Stateless
  - Router vs. Bridge
  - Configuration/Security model

Packet Filtering

- ✔ Acts like a router or bridge
  - Does not modify network connections or packet headers
- ✔ Allow/Deny packets based on packet data
- ✔ Allow/Deny packets based on Input/Output interface
  - shorthand for source or destination

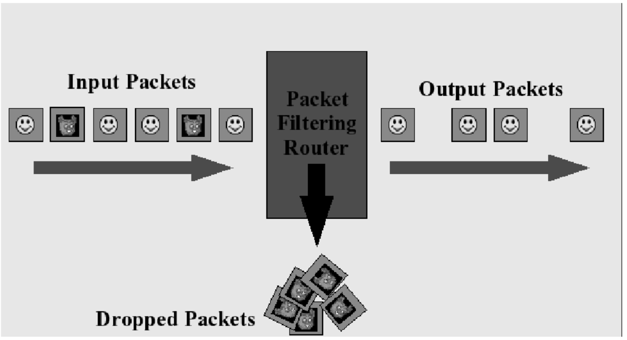
Allow/Deny packets based on packet data:

- ✔ Layer 2:
  - Source or Destination MAC addresses
- ✔ Layer 3:
  - Source or Destination addresses, ports
  - Protocol or Protocol details
    - ex: disallow IP Source Routing
    - disallow ICMP redirect packets
    - disallow common “malicious” packet signatures

Allow/Deny packets based on packet data:

- ✔ Layer 4:
  - Service-specific (ex: by URL)

Packet Filtering



Packet Filtering Rules

- ✔ Typically applied in a specific order
  - First match applies
- ✔ One filter per rule
- ✔ Default rule?
  - “Default Deny” safest
  - Warning: implied default rule: Deny or Allow?

Example Packet Filtering Rules:

- ✔ Protect 128.105.0.0 network with Cisco router access control lists
- ✔ Apply rules from top to bottom:

```
deny   ip   128.105.0.0 0.0.255.255 any
permit tcp  any 128.105.1.1 eq 25
permit tcp  any 128.105.1.2 eq 80
permit tcp  any 128.105.1.3 eq 22
deny   icmp any any redirect log
permit icmp any 128.105.1.4 echo
deny   icmp any any echo log
deny   ip   any any log
```

Example Packet Filtering Rules:

- ✔ Protect 128.105.0.0 network with OpenBSD pf:

```
block in log all
block in log quick on $campus_if from
    128.105.0.0/16 to any
pass in quick on $campus_if proto tcp
    from any to 128.105.1.1/32 port = 25
...
pass in quick on $cs_if proto tcp from
    128.105.0.0/16 to any keep state
```

Packet Filtering Advantages

- ✔ Can be placed at a few “strategic” locations
  - Internet/Internal network border router
  - To isolate critical servers
- ✔ Efficient
- ✔ Simple concept

Packet Filtering Advantages

- ✔ Widely available
  - Implemented in most routers
  - Firewall appliances
  - Open Source operating systems and software
  - Specialized network interface cards with filtering capabilities
    - Download up to 64k rules to some

Packet Filtering Disadvantages

- ✔ Hard to configure
  - – Rules can get complex
- ✔ Hard to test and verify rules
- ✔ Incomplete implementations
- ✔ Bugs often “fail unsafe” -- allow unintended traffic to pass

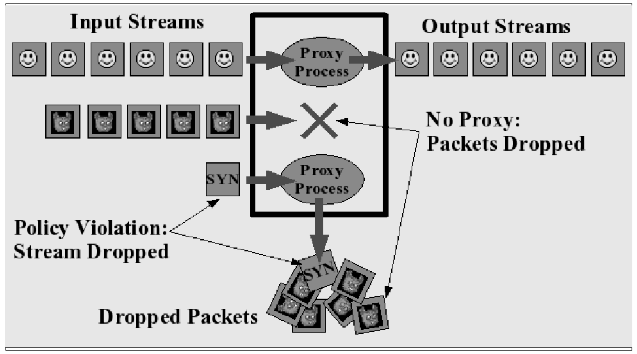
Packet Filtering Disadvantages

- ✔ Can Reduce router performance
- ✔ Some policies don't map to packet filtering

Proxy Firewalls

- ✔ Specialized application to handle specific traffic
- ✔ Protocol gateways
  - – Creates new network connection, forwards data between “inside” and “outside” connection
- ✔ May apply service-specific rules & policies

Proxy Firewall



Proxy Advantages

- ✔ Can do "intelligent" filtering
- ✔ Can perform user-level authentication
- ✔ Can use information from outside the connection or packet stream
- ✔ Can protect weak/faulty IP implementations
  - Separate network connections to source, destination

Proxy Advantages

- ✔ Can provide application/service-specific services or actions:
  - data caching
  - data/connection logging
  - data filtering/selection or server selection based on source/destination or other status visible to proxy
  - add or apply routing/bandwidth policy

Proxy Disadvantages

- ✔ Need to write/install proxy for each service
  - Lag time to develop proxy for new service
- ✔ May need dedicated proxy servers for each service
- ✔ Often need cooperation of clients, servers

Dealing with Connections

- ✔ Typical scenario:
  - Restrict incoming connections to specific services and servers
    - Allow traffic to public web site
    - Allow inbound e-mail to mail gateway
  - Allow unlimited outgoing connections
    - Employees can browse the web, send e-mail, etc
    - Firewall needs to track connections to do this

TCP Connections

- ✔ Outbound new connections often from dynamic (unpredictable) src port
  - Can't use firewall rule based on src port
- ✔ Destination may be "well-known" port
  - But not always
- ✔ Destination may move to dynamic port during connection establishment

TCP Connection Setup

SRC PORT: ABC
DST PORT: 25
SYN

SRC PORT: XYZ
DST PORT: ABC
SYN ACK

SRC PORT: ABC
DST PORT: XYZ
ACK

SRC PORT: XYZ
DST PORT: ABC
ACK

UDP "Connections"

- ✔ UDP is stateless
- ✔ "Connection" or "Session" implied by one or more packets from SRC to DST, one or more packets back
  - May or may not be on "well-known" port
  - May or may not be on same port as original traffic

UDP Session: DNS Query

SRC PORT: ABC
DST PORT: 53

SRC PORT: XYZ
DST PORT: ABC

SRC PORT: XYZ
DST PORT: ABC

Handling TCP Connections Without State

✔ How to detect "established" TCP connections without keeping state?

- Established connections have ACK flag set

✔ "Established" keyword in many stateless firewalls allows incoming packets if ACK flag set

- Can be exploited by faking packets with ACK flag set

UDP Connections Without State

✔ Can't be done - not enough information in each packet

Keeping State

✔ Stateless firewalls easy to implement

- memory/CPU requirements are low
- no routing impact
- but can only act on information from the packet

Keeping State

✔ Statefull/Dynamic firewalls have more information to use in decision making

- Keeping state is more complicated

✔ Proxy Firewalls often keep state

- But packet filtering firewalls can be statefull too

Using State Information: TCP

✔ Keep Track of outbound TCP packets:

- If match on existing “session”, update session data
- If session setup packet (SYN, no ACK), create new session in state table
  - keep until session ended
- If session shutdown packet
  - delete session from state table

Using State Information: TCP

✔ Inbound TCP packets:

- match to existing TCP session: allow packet
- Otherwise, reject packet

✔ Track TCP session state, delete session from state table when finished

Using State Information: UDP

✔ Keep track of outbound UDP packets:

- If match on existing “session”, update session data
- Otherwise, create new “session” in state table
  - Keep session state for some time interval

✔ Inbound UDP packets:

- Match to existing “session” -> allow packet
- Otherwise, reject packet

Using State Information: UDP

✔ Only works for typical same-port scenario

- Reply must come from same IP as outbound traffic, go to same IP and port as outbound traffic

✔ More complicated session-setup protocols won't work

Distributed Firewalls

- ✔ 2 or more firewalls
  - share the load
  - redundancy in event of hardware or routing failure
- ✔ Need to keep rules synchronized
- ✔ Need to keep state synchronized
  - Asymmetric routes will cause connection drops without fully synchronized state

Routing Firewalls

- ✔ Most firewalls act as routers
- ✔ Each interface has an IP address
- ✔ Packet processing:
  - Filters applied
  - IP stack traversed
    - TTL decremented
    - Packet routed for delivery to destination

Routing Firewalls

- ✔ Visible in network
- ✔ Needs to be in routing table of immediate neighbors
- ✔ Shows in traceroute

Bridging Firewalls

- ✔ "Bump in the road"
- ✔ Interfaces do not have IP addresses
- ✔ Packet processing:
  - Filters applied
  - No IP stack in firewall path
    - IP TTL NOT decremented
  - Packet forwarded towards destination

Bridging Firewalls

- ✔ Not visible in network
- ✔ No changes in neighbor configuration
- ✔ Not visible in traceroute
- ✔ Debugging more difficult

Internal Firewalls

- ✔ Gaining popularity in larger organizations
- ✔ Not safe to assume that all “bad guys” are outside
- ✔ Prevent accidents, isolate damage
- ✔ Apply appropriate security policies to selected servers/areas of operation

Internal Firewalls

- ✔ Separate internal operations should be isolated on the network
  - Example: Purchasing and Accounts Payable
  - Different parts of the organization have relationships with different outside groups
  - Outside groups may be competitors, require isolation from each other

Related Technologies

- ✔ Network Address Translation

Network Address Translation

✔ Specialized proxy

- Rewrites IP addresses, ports
- Map "private" IP addresses to "public" addresses
  - Conserve IP address space
  - RFC 1918
- Virtual servers, load balancing

Network Address Translation

✔ Protects unmapped "inside" addresses

- not visible at all to "outside" addresses

Network Address Translation

✔ Implemented in most home "broadband" routers

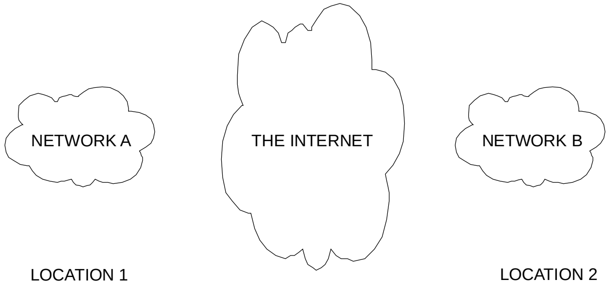
- 1 IP address from broadband network
- multiple computers and IP addresses "inside" home network
- limited capability to specify "inside" addresses/ports to expose to "outside"
- usually includes a limited firewall capability

Virtual Private Networks (VPNs)

Virtual Private Networks (VPNs)

- ✔ Tunnel traffic from host/network A to host/network B
  - – Encapsulate in another protocol (IP, SSH, etc)
  - – Usually includes encryption, authentication
- ✔ Block all external traffic except to "public" services
- ✔ Allow only VPN traffic to internal services

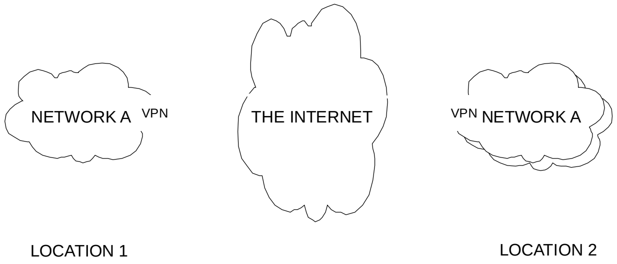
Two Locations, Two Networks



What We Want

NETWORK A
NETWORK A
LOCATION 1
LOCATION 2

Using a VPN



Virtual Private Networks (VPNs)

- Danger: VPN traffic usually bypasses firewall…
- VPN can allow “outside” traffic to bypass firewall
  - Other systems at home/remote location may incorrectly route via VPN
- Can lower the “inside” security standard
  - Other remote systems may not be patched…

Network Intrusion Detection Systems

Network Intrusion Detection Systems

- Security model
- Types of IDS systems

NIDS Security Model

- Analyze live network traffic, attempt to detect malicious traffic
  - Raise an alert (common)
  - Reconfigure firewall in “real time” to block malicious traffic (not common)
- Log traffic or signatures for later analysis

Types of NIDS

- ✔ Signature based systems
- ✔ Learning systems

Signature-based NIDS

- ✔ Most NIDS use signatures
- ✔ Like virus detection systems
- ✔ Pattern-match traffic against known signatures (patterns) of “bad” traffic
  - Lag in identifying signatures of new attacks
  - May need a new signature for each variant/implementation of an attack

Signature-based NIDS

- ✔ Limitations of signature descriptions/matching limit effectiveness
- ✔ Most systems/signatures only examine individual packets
  - Stateless
- ✔ Some systems consider multiple packets
  - Rate, multi-packet pattern-match, ...

Additional NIDS Features

- ✔ Vary by implementation:
  - Database support
  - Logging capabilities
  - Bandwidth limitations
  - Distributed Sensors
  - Alert generation
  - Report generation

Example: SNORT

- ✔ Open Source Network Intrusion Detection System
- ✔ Mostly signature-based, also includes many additional methods via plug ins
- ✔ Over 2,000 rules developed by the SNORT community

Example SNORT Rule: "BackOrifice" access attempt

```
alert tcp $HOME_NET 80 ->
 $EXTERNAL_NET any (msg:"BACKDOOR
 BackOrifice access"; flags: A+;
 content: "server|3a| BO|2f|";
 reference:arachnids,400; sid:112;
 classtype:misc-activity; rev:3;)
```

Example SNORT Rule: "UDP ECHO+Chargen Bomb"

```
alert udp any 19 <> any 7 (msg:"DOS
 UDP echo+chargen bomb";
 reference:cve,CAN-1999-0635;
 reference:cve,CVE-1999-0103;
 classtype:attempted-dos; sid:271;
 rev:3;)
```

Example SNORT Rule: X86 Linux samba overflow

```
alert tcp $EXTERNAL_NET any ->
 $HOME_NET 139 (msg:"EXPLOIT x86
 Linux samba overflow";
 flow:to_server,established;
 content:"|eb2f 5feb 4a5e 89fb
 893e 89f2|"; reference:bugtraq,
 1816;
 reference:cve,CVE-1999-0811;
 reference:cve,CVE-1999-0182;
 classtype:attempted-admin; sid:
 292; rev:5;)
```

“Learning” NIDS

- Idea: Use AI techniques to “learn” about expected (good) traffic
  - Anything else is a potential attack
- Mostly still a research topic
- Hard to provide accurate training data
  - How do you know there isn't an attack in progress during the “normal” training?

NIDS Strengths

- Organized way to analyze traffic
- Can detect attacks, policy violations, mis configured systems

NIDS Weaknesses

- Potential for many false positives
  - ex: CS “mirror” server
    - every linux distribution includes files with “dangerous” assembly language sequences (the boot loader, trap handler, etc)
    - NIDS detect packets downloading those files...
  - ex: SNORT at CS border reported thousands of potential attacks every day

NIDS Weaknesses

- Hard to distinguish between attempted attack and successful attack
  - Requires keeping state
  - Requires more sophisticated signature definitions and matching tools
- Need to customize rule set to each site
- Need to keep rule set up-to-date with current vulnerabilities and attacks ...

Internet Sinks and Honeypots

✔ Divert Internet traffic to another system

  - Blackhole/Sinkhole routers
  - Tarpits

✔ Honeypots: “fake” hosts that look vulnerable

✔ Goal: capture attack/intrusion traffic for analysis

Coordinated Anomaly and Intrusion Detection

✔ Research by Professor Barford and others

✔ Global coordinated intrusion detection infrastructure

  - Combining multi-site data from firewalls, NIDS, and Internet Sinks

✔ Goal: Decrease reaction time to new worm outbreaks, reduce false alarm rates, and automatically generate counter measures

Port Scanning

Port Scanning

✔ “Bad guys” scan networks for open network ports to exploit

✔ Same technique can be used to assess/test a network

Port Scanning

- ✔ Simple: attempt connection to each TCP, UDP port
- ✔ More complex: send protocol-specific traffic to each port
  - Identify implementation of service by response
  - Identify/attempt to exploit specific vulnerabilities

Port Scanning

- ✔ nmap
- ✔ Nessus
- ✔ Commercial port scanners

Network Management

Network Management

- ✔ Good network management methods increase network security
  - Monitor bandwidth usage
  - Detect excessive/unexpected traffic surges
- ✔ Tools for rapid traffic isolation

Network Management

✔ Tools to identify source/destination of traffic

- Which computer is causing a traffic surge?
- Physical location as well as IP address

✔ Tools for rapid reconfiguration of network devices (switches, routers, etc)

✔ Keep network device firmware/software up-to-date

Configuration Management

Configuration Management

✔ Good system administration methods increase network security

- Only configure network services where needed
  - Turn off unneeded, potentially vulnerable services on most computers
- Automate installation & configuration of computers on network

Configuration Management

✔ Tools to audit computer configurations

- Know use/purpose of each computer
- Verify correct configuration of each computer

✔ Apply latest OS and application patches

- Tools to rapidly deploy patches

✔ Organized computer deployment will allow for better firewall deployment

CSL Network Security

Computer Systems Lab Network Security

- ✔ CSL supports all CS Department computing
  - Instructional, research, administrative
  - Manage CS network
- ✔ Integrated staff:
  - Windows, Unix, Network, Hardware, etc...
  - Some specialization, all on same team
  - Everyone involved in security

CS Firewalls: Our Method

- ✔ "Insiders" are generally more trustworthy than "outsiders"
  - But sometimes "bad guys" get in – stolen passwords, unhappy students, etc
- ✔ Divide computers by level of threat, level of security available

CS Firewalls: Our Method

- ✔ Multi-layer firewall for special networks:
  - Border firewall
  - Firewall or Router closest to the network
- ✔ Try and keep out of the way of legitimate users:
  - CS researchers do unexpected things
  - default "allow"

CS Border Firewall

- ✔ "Trip Curb"
  - You can stub your toe if you kick it
  - Rules getting more complex... the curb is taller and more solid now
  - 211 rules: 125 block, 86 pass, 466 lines total
- ✔ Screening/Packet Filtering firewall
  - Statefull
  - OpenBSD bridging firewall

CS Border Firewall: Input Rules

- ✔ Default "allow"
  - Block known problem ports
  - Block unneeded services with potential problems
    - NFS, RPC, NETBIOS ...
- ✔ Block forged/malformed packets
  - Inbound with our SRC address
  - Inbound with "unroutable" SRC addresses

CS Border Firewall: Input Rules

- ✔ Enforce some policies
  - SMTP only to mail gateways (virus scanning)
  - WWW only to known web servers
- ✔ Allow inbound packets for established connections/sessions (statefull)
- ✔ Block all traffic to special networks

CS Border Firewall: Output Rules

- ✔ Block forged/malformed packets
  - Outbound without our SRC address
- ✔ Block all traffic from special networks

CS Border Firewall: Next Steps

- ✔ Switch to “default deny”
- ✔ Better analysis tools

Other CS Firewalls

- ✔ Unpatched/Experimental network
  - Can only reach other CS networks
  - Can not send/receive email (even inside CS)
- ✔ Crash-and-Burn network
  - Can only reach other CS networks
  - Some services restricted

Other CS Firewalls

- ✔ Wireless/Laptop network
  - Can only do DNS until authenticated
- ✔ Install network
  - Used by CSL for installing OS on new computers
  - Isolated from internet to prevent attacks before OS installation/patching complete

Other CS Firewalls

- ✔ Printer network
  - Most printers run un-patchable/insecure software
    - including a web server for configuration & status
  - Only allow access to print servers from CS
  - Only allow access to printers from print servers

Other CS Firewalls

- ✔ Network maintenance network:
  - Administrative access to switches and routers
  - Restricted to admin networks
- ✔ Host firewalls
  - Second layer of defense
  - Isolate VMware virtual networks from production network

CS Network Intrusion Detection

- ✔ Deployed SNORT at network border
- ✔ With default rules, thousands of events logged every day

SNORT Events

- ✔ With modified rules, thousands of events logged every day
- ✔ Many port scans every day
- ✔ Many intrusion attempts every day
  - Not vulnerable:
  - Wrong OS, IP not in use, port not open, firewall, service patched, ...

CS Network Intrusion Detection

- ✔ Need better way to filter reports
- ✔ Very useful in finding problems
- ✔ Very labor intensive: need better tools
- ✔ Currently not active (lack of staff)

CSL Port Scanning

- ✔ Participated in research project to develop "state-of-the-art" security audits
- ✔ Project initiated regular, systematic network vulnerability scanning
  - Nessus
- ✔ Very effective at finding vulnerabilities and configuration problems

CSL Port Scanning

- ✔ Very labor intensive
- ✔ Need better tools
- ✔ Very effective when combined with other tools:
  - Firewalls
  - NIDS
    - cross-reference intrusion alerts to known vulnerabilities, known "safe" hosts

CSL Network Management

- ✔ Active management of the network
- ✔ Active monitoring of network traffic, errors, etc
- ✔ Switch ports "MAC-locked" to specific interface
  - coordinated with inventory and configuration management system

CSL Network Management

- ✔ Switch ports "MAC-locked"
  - restricted to MAC address of assigned computer
  - Prevent "bandwidth borrowers"
  - Prevent rouge computers on our network
  - Not perfect: MAC addresses can be reset on most ethernet cards

CSL Configuration Management

- ✔ All "production" computers actively managed by CSL
- ✔ Good tools for patch deployment, configuration verification

Questions?