Lecture 24
November 22, 2024

Firewalls

- Mediate access to organization’s network
  - Also mediate access out to the Internet
- Example: Java applets filtered at firewall
  - Use proxy server to rewrite them
    - Change “<applet>” to something else
  - Discard incoming web files with hex sequence CA FE BA BE
    - All Java class files begin with this
  - Block all files with name ending in “.class” or “.zip”
    - Lots of false positives

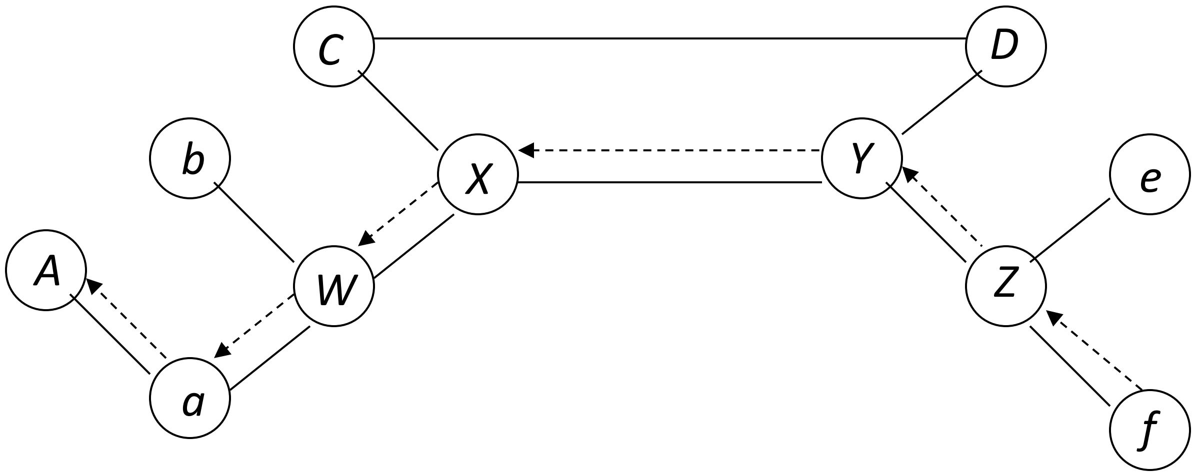
Intrusion Detection and Isolation Protocol

- Coordinates reponse to attacks
- *Boundary controller* is system that can block connection from entering perimeter
  - Typically firewalls or routers
- *Neighbor* is system directly connected
- *IDIP domain* is set of systems that can send messages to one another without messages passing through boundary controller

Protocol

- IDIP protocol engine monitors connection passing through members of IDIP domains
  - If intrusion observed, engine reports it to neighbors
  - Neighbors propagate information about attack
  - Trace connection, datagrams to boundary controllers
  - Boundary controllers coordinate responses
    - Usually, block attack, notify other controllers to block relevant communications

Example



- *C*, *D*, *W*, *X*, *Y*, *Z* boundary controllers
- *f* launches flooding attack on *A*
- Note after *X* suppresses traffic intended for *A*, *W* begins accepting it and *A*, *b*, *a*, and *W* can freely communicate again

Follow-Up Phase

- Take action external to system against attacker
  - Thumbprinting: traceback at the connection level
  - IP header marking: traceback at the packet level
  - Counterattacking

Thumbprinting

- Compares contents of connections to determine which are in a chain of connections
- Characteristic of a good thumbprint
  1. Takes as little space as possible
  2. Low probability of collisions (connections with different contents having same thumbprint)
  3. Minimally affected by common transmission errors
  4. Additive, so two thumbprints over successive intervals can be combined
  5. Cost little to compute, compare

Example: Foxhound

- Thumbprints are linear combinations of character frequencies
  - Experiment used *telnet*, *rlogin* connections
- Computed over normal network traffic
- Control experiment
  - Out of 4000 pairings, 1 match reported
    - So thumbprints unlikely to match if connections paired randomly
    - Matched pair had identical contents

Experiments

- Compute thumbprints from connections passing through multiple hosts
    - One thumbprint per host
- Injected into a collection of thumbprints made at same time
    - Comparison immediately identified the related ones
- Then experimented on long haul networks
    - Comparison procedure readily found connections correctly

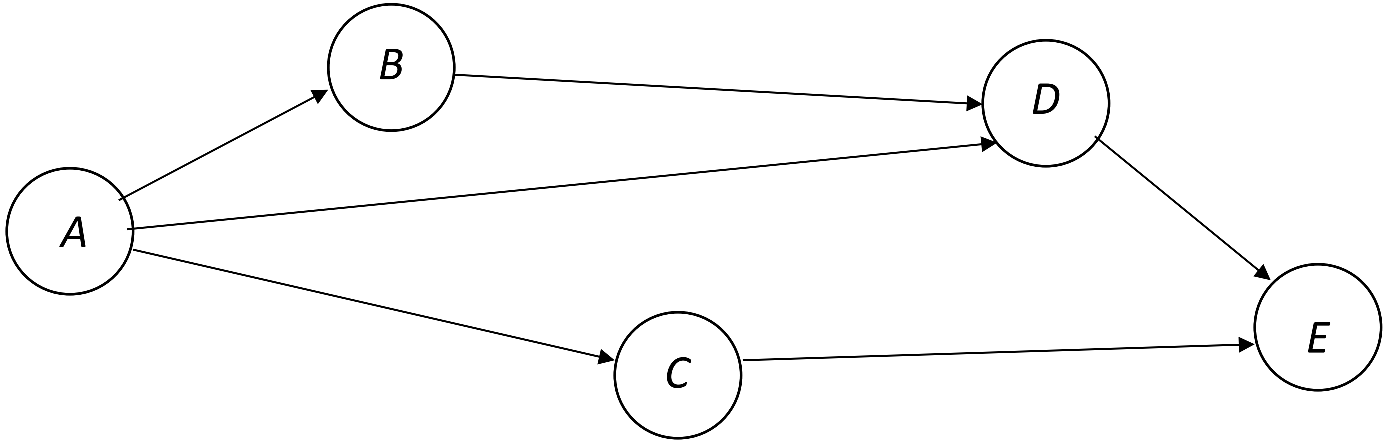
IP Header Marking

- Router places data into each header indicating path taken
- When do you mark it?
  - Deterministic: always marked
  - Probabilistic: marked with some probability
- How do you mark it?
  - Internal: marking placed in existing header
  - Expansive: header expanded to include extra space for marking

Example: Probabilistic Scheme

- Expand header to have *n* slots for router addresses
- Router address placed in slot *s* with probability *sp*
- Use: suppose SYN flood occurs in network

Use



- *E* SYN flooded; 3150 packets could be result of flood
- 600 (*A*, *B*, *D*); 200 (*A*, *D*); 150 (*B*, *D*); 1500 (*D*); 400 (*A*, *C*); 300 (*C*)
  - *A*: 1200; *B*: 750; *C*: 700; *D*: 2450
- Note traffic increases between *B* and *D*
  - *B* probable culprit

Algebraic Technique

- Packets from A to B along path P
- First router labels jth packet with x_j
- Routers on P have IP addresses $a_0, \ldots, a_n$
- Each router a_i computes $Rx_j + a_i$, R being current mark $a_0 x_j^i + \ldots + a_{i-1}$ (Horner's rule)
  - At B, marking is $a_0 x^n + \ldots + a_n$, evaluated at x_j
- After n+1 packets arrive, can determine route

Alternative

- Alternate approach: at most *l* routers mark packet this way
- *l* set by first router
- Marking routers decrement it
- Experiment analyzed 20,000 packets marked by this scheme; recovered paths of length 25 about 98% of time

Problem

- Who assigns x_j?
  - Infeasible for a router to know it is first on path
  - Can use weighting scheme to determine if router is first
- Attacker can place arbitrary information into marking
  - If router does not select packet for marking, bogus information passed on
  - Destination cannot tell if packet has had bogus information put in it

Counterattacking

- Use legal procedures
  - Collect chain of evidence so legal authorities can establish attack was real
  - Check with lawyers for this
    - Rules of evidence very specific and detailed
    - If you don't follow them, expect case to be dropped
- Technical attack
  - Goal is to damage attacker seriously enough to stop current attack and deter future attacks

Consequences

1. May harm innocent party
   - Attacker may have broken into source of attack or may be impersonating innocent party
2. May have side effects
   - If counterattack is flooding, may block legitimate use of network
3. Antithetical to shared use of network
   - Counterattack absorbs network resources and makes threats more immediate
4. May be legally actionable

Counterattacking

- Use legal procedures
  - Collect chain of evidence so legal authorities can establish attack was real
  - Check with lawyers for this
    - Rules of evidence very specific and detailed
    - If you don’t follow them, expect case to be dropped
- Technical attack
  - Goal is to damage attacker seriously enough to stop current attack and deter future attacks

Consequences

1. May harm innocent party
   - Attacker may have broken into source of attack or may be impersonating innocent party
2. May have side effects
   - If counterattack is flooding, may block legitimate use of network
3. Antithetical to shared use of network
   - Counterattack absorbs network resources and makes threats more immediate
4. May be legally actionable

Example: Counterworm

- Counterworm given signature of real worm
  - Counterworm spreads rapidly, deleting all occurrences of original worm
- Some issues
  - How can counterworm be set up to delete *only* targeted worm?
  - What if infected system is gathering worms for research?
  - How do originators of counterworm know it will not cause problems for any system?
    - And are they legally liable if it does?

Incident Response Groups

- *Computer security incident response team* (CSIRT): team established to assist and coordinate responses to a security incident among a defined constituency
  - “Constituency” defined broadly; may be vendor, company, sector such as financial or academic, nation, etc.
- Mission depends in large part on constituency
  - Critical part: keep constituency informed of services CSIRT provides, how to communicate with CSIRT

Example: CERT/CC

- Grew out of Internet worm, when many groups dealt with it and had to communicate with one another
  - In some cases, they did not know about other groups, what they are doing
  - Sometimes trusted third party did introduction
- Raised concerns of how to communicate and coordinate responses to future events
- Led to development of Computer Emergency Response Team (CERT, later CERT/CC)

CSIRT Missions

1. *Publication*: publish policies, procedures about what it can do, how it will communicate with constituency, how constituency can communicate it
2. *Collaboration*: collaborate with other CSIRTs to gather, disseminate information about attacks, respond to attacks
3. *Secure communication*: preserve credibility; ensure constituency they are communicating with CSIRT and not masquerader; and CSIRT must be sure it is dealing with affected members of constituency and other CSIRTs, not masqueraders

How a CSIRT Functions

- Policy defines what it will, will not do
- Plan how to respond to incidents, driven by needs and constraints of constituents
  - Avoid solely technical approach
  - Couple that with strategic analysis to find organizational issues contributing to attack or hindering appropriate responses
  - Understanding incident involves non-technical aspects of organization such as people, resources, economics, laws and regulations
- Disseminate information to prevent, limit attacks
  - Include vulnerability reports

Digital Forensics

The science of identifying and analyzing entities, states, state transitions of events that have occurred or are occurring

- Also called *computer forensics*
- Usually done to figure out what caused an anomaly or understand nature of attack: how did attackers (try to) enter system, what they did, and how defenses failed
- *Legal forensics* may include digital forensics
  - Here, analysts must acquire information and perform analysis in such a way that what is uncovered can be admitted into a legal proceeding

Goals of Forensics Principles

- *Locard's Exchange Principle*: every contact leaves a trace
- Forensics principles create environment in which Locard's Exchange Principle holds
- Must consider entire system
  - Attack on one component may affect other components
  - Multistage attacks leverage compromise of a component to compromise another
  - Attack may have effects that analyst does not expect

Principle 1: Consider the Entire System

- Analyst needs access to information the intruder had before, after attack
  - Includes changes to memory, kernel, file systems, files
- Rarely recorded continuously, so information incomplete
- Logs also often omit useful information
  - Record connections, states of connections, services, programs executed
  - Omit directories searched to find dynamically loaded libraries, or which ones are loaded; also omit memory contents during program execution
  - Application logging also may not log security-relevant events

Principle 2: Assumptions Should Not Control What Is Logged

- Analysts work from logs capturing information before, during, after incident being analyzed
  - If assumptions guide what is being logged, information may be incomplete
- Record enough information to reconstruct system state at any time
  - Virtual machine introspection great for this

Example: ExecRecorder

Architecture to enable replay of events with minimal overhead and no changes to operating system

- Hypervisor Bochs contains checkpoint, logging, replay mechanisms
  - These are invisible to operating system running in Bochs
- Checkpoint component takes snapshots of system state
- Logging component records nondeterministic events to enable them to be reproduced *exactly*
- Replay component reconstructs and restores state of system, and system activity occurs from that point on

Principle 3: Consider the Effects of Actions As Well As the Actions

- Aim is to establish what system did as well as what attacker did
- Logs record actions, sometimes effects, but almost never causes allowing actions to occur
- Example: remote attacker gains enough access to execute commands on other systems
  - Logs show which server she went to, commands issued
  - Logs do not show vulnerability that enabled attacker to succeed, so others may exploit the same vulnerability

Principle 4: Context Assists in Understanding Meaning

- Same action may cause 2 different effects when executed in 2 different contexts
- Example: LINUX command typed at keyboard (not full path name of command)
  - What gets executed depends on search path, contents of file system
- Example: file system monitoring tool logging access to files by file name
  - The same name may refer to 2 different files (refers to file X, then file X deleted and a new file X created)

Principle 5: Information Must Be Processed, Presented in an Understandable Way

- Those who need to understand the forensic analysis can do so
- First audience: analysts
  - Interfaces to forensic tools must be designed with usability in mind, and indicate where gaps in data, analysis are
  - Presentation of results must also be clear to a technical audience
- Second audience: non-technical audience
  - Provide information in a way that the audience can understand what happened, how it happened, what the effects of the attack were, the level of assurance that the data, analysis is correct
  - May need to present evidence in a way appropriate to a particular audience, such as legal audiences

Practice

Typically 4 steps to reconstruct state of system and sequence of actions of interest

1. Capture, preserve current state of system, network data
2. Extract information about that state and prior states
   - Reverse these steps if system is active; in this case, state will be approximate as gathering data takes time and state may change during that process
3. Analyze data to determine sequence of actions, objects affected, and how they are affected
4. Prepare, report results of analysis to intended audience

Gathering Data

- Get a complete image of all components
- If infeasible (because compromise discovered after it is done, or system is active), get as complete an image as possible
  - May include disk images, backups, stored network or IDS data
- Be sure to make cryptographic hash of all data
  - That way, you and others can verify data is unaltered after being checksummed

Example: Gathering Data

- Disk is full, but space used by files much less than size of disk
- Sysadmin removes disk, mounts it read-only on another system
- Sysadmin creates image of it on some other media
  - On a second, previously wiped, disk
- Sysadmin creates cryptographic checksum of image
  - Can be used to show image was not changed since its creation
- Sysadmin uses a different program to recompute checksum and verifies it matches previously computed checksum
  - Used to ensure cryptographic checksum is correct

Persistent vs. Volatile Data

- Persistent data: remains when system or data storage is powered off
  - Data on hard drive or secondary storage
- Volatile data: transient, disappearing at some point in time (like when system is powered off)
  - Data in memory
  - More difficult to capture than persistent data