Intrusion Detection
Computer Security
Peter Reiher
February 11, 2021

Outline

- Introduction
- Characteristics of intrusion detection systems
- Some sample intrusion detection systems

Introduction

- Many mechanisms exist for protecting systems from intruders
  - Access control, firewalls, authentication, etc.
- They all have one common characteristic:
  - *They don't always work*

Intrusion Detection

- Work from the assumption that sooner or later your security measures will fail
- Try to detect the improper behavior of the intruder who has defeated your security
- Inform the system or system administrators to take action

Why Intrusion Detection?

- If we can detect bad things, can't we simply prevent them?
- Possibly not:
  - May be too expensive
  - May involve many separate operations
  - May involve things we didn't foresee

For Example,

- Your intrusion detection system regards setting uid on root executables as suspicious
  - Yet the system must allow the system administrator to do so
- If the system detects several such events, it becomes suspicious
  - And reports the problem

Couldn't the System Just Have Stopped This?

- Perhaps, but -
- The real problem was that someone got root access
  - The changing of setuid bits was just a symptom
- And under some circumstances the behavior is legitimate

Intrusions

- "any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource"[1]
- Which covers a lot of ground
  - Implying they're hard to stop

[1]Heady, Luger, Maccabe, and Servilla, "The Architecture of a Network Level Intrusion Detection System," Tech Report, U. of New Mexico, 1990.

Kinds of Intrusions

- External intrusions
- Internal intrusions

External Intrusions

- What most people think of
- An unauthorized (usually remote) user trying to illicitly access your system
- Using various security vulnerabilities to break in
- The typical case of a hacker attack

Internal Intrusions

- An authorized user trying to gain privileges beyond those he should have
- Used to be most common case
- No longer the majority of problems
  - But often the most serious ones
- More dangerous, because insiders have a foothold and know more

Information From 2016 Verizon Report[1]

- Data from dozens of major cybersecurity organizations, covering 100,000+ breaches
- Indicates external breaches in around 80% of cases
- Insider attack components in around 15% of all cases
  - Some involved both insiders and outsiders

[1] http://www.verizonenterprise.com/verizon-insights-lab/dbir/2016/

Basics of Intrusion Detection

- Watch what's going on in the system
- Try to detect behavior that characterizes intruders
- While avoiding improper detection of legitimate access
- At a reasonable cost

Intrusion Detection and Logging

- A natural match
- The intrusion detection system examines the log
  - Which is being kept, anyway
- Secondary benefits of using the intrusion detection system to reduce the log

On-Line Vs. Off-Line Intrusion Detection

- Intrusion detection mechanisms can be complicated and heavy-weight
- Perhaps better to run them off-line
  - E.g., at nighttime
- Disadvantage is that you don't catch intrusions as they happen

Failures In Intrusion Detection

- False positives
  - Legitimate activity identified as an intrusion
- False negatives
  - An intrusion not noticed
- Subversion errors
  - Attacks on the intrusion detection system

Desired Characteristics in Intrusion Detection

- Continuously running
- Fault tolerant
- Subversion resistant
- Minimal overhead
- Must observe deviations
- Easily tailorable
- Evolving
- Difficult to fool

Host Intrusion Detection

- Run the intrusion detection system on a single computer
- Look for problems only on that computer
- Often by examining the logs of the computer

Advantages of the Host Approach

- Lots of information to work with
- Only need to deal with problems on one machine
- Can get information in readily understandable form

Network Intrusion Detection

- Do the same for a local (or wide) area network
- Either by using distributed systems techniques
- Or (more commonly) by sniffing network traffic

Advantages of Network Approach

- Need not use up any resources on users' machines
- Easier to properly configure for large installations
- Can observe things affecting multiple machines

Network Intrusion Detection and Data Volume

- Lots of information passes on the network
- If you grab it all, you will produce vast amounts of data
- Which will require vast amounts of time to process

Network Intrusion Detection and Sensors

- Use programs called *sensors* to grab only relevant data
- Sensors quickly examine network traffic
  - Record the relevant stuff
  - Discard the rest
- If you design sensors right, greatly reduces the problem of data volume

Network Intrusion Detection and Deep Packet Inspection

- An obvious situation to do deep packet inspection
- Only looking at headers usually won't cut it
- But deep packet inspection is expensive
- So vital to use sensors to choose where to go deep
  - Only do serious analysis on small percentage of packets

Wireless IDS

- Observe behavior of wireless network
  - Generally 802.11
- Look for problems specific to that environment
  - E.g., attempts to crack WEP keys
- Usually doesn't understand higher network protocol layers
  - And attacks on them

Application-Specific IDS

- An IDS system tuned to one application or protocol
  - E.g., SQL
- Can be either host or network
- Typically used for machines with specialized functions
  - Web servers, database servers, etc.
- Possibly much lower overheads than general IDS systems

Styles of Intrusion Detection

- Misuse intrusion detection
  - Try to detect things known to be bad
- Anomaly intrusion detection
  - Try to detect deviations from normal behavior
- Specification intrusion detection
  - Try to detect deviations from defined "good states"

Misuse Detection

- Determine what actions are undesirable
- Watch for those to occur
- Signal an alert when they happen
- Often referred to as *signature detection*

Level of Misuse Detection

- Could look for specific attacks
  - E.g., SYN floods or IP spoofing
- But that only detects already-known attacks
- Better to also look for known suspicious behavior
  - Like trying to become root
  - Or changing file permissions

How Is Misuse Detected?

- By examining logs
  - Only works after the fact
- By monitoring system activities
  - Often hard to trap what you need to see
- By scanning the state of the system
  - Can't trap actions that don't leave traces
- By sniffing the network
  - For network intrusion detection systems

Pluses and Minuses of Misuse Detection

+ Few false positives
+ Simple technology
+ Hard to fool
  - At least about things it knows about

– Only detects known problems

– Gradually becomes less useful if not updated

– Sometimes signatures are hard to generate

Misuse Detection and Commercial Systems

- Essentially all commercial intrusion detection systems primarily detect misuse
  - Generally using signatures of attacks
- Many of these systems are very similar
  - Differing only in details
- Differentiated primarily by quality of their signature library
  - How large, how quickly updated

Anomaly Detection

- Misuse detection can only detect known problems
- And many potential misuses can also be perfectly legitimate
- Anomaly detection instead builds a model of valid behavior
  - And watches for deviations

Methods of Anomaly Detection

- Statistical models
  - User behavior
  - Program behavior
  - Overall system/network behavior
- Expert systems
- Pattern matching of various sorts
- Misuse detection and anomaly detection sometimes blur together
- Machine-learning based

Pluses and Minuses of Anomaly Detection

+ Can detect previously unknown attacks
+ Not deceived by trivial changes in attack
– Hard to identify and diagnose nature of attacks
– Unless careful, may be prone to many false positives
– Depending on method, can be expensive and complex

Anomaly Detection and Academic Systems

- Most academic research on IDS in this area
  - More interesting problems
  - Greater promise for the future
  - Increasingly, misuse detection seems inadequate
  - Lots of machine-learning approaches here
- But few really effective systems currently use it
  - Not entirely clear that will ever change
  - What if it doesn't?

Specification Detection

- Define some set of states of the system as good
- Detect when the system is in a different state
- Signal a problem if it is

How Does This Differ From Misuse and Anomaly Detection?

- Misuse detection says that certain particular things are bad
- Anomaly detection says deviations from statistically normal behavior are bad
- Specification detection defines exactly what is good and calls the rest bad

Some Challenges

- How much state do you have to look at?
  - Typically dealt with by limiting observation to state relevant to security
  - Easy to underestimate that . . .
- How do you specify a good state?
- How often do you look?
  - Might miss attacks that transiently change the state

Protocol Anomaly Detection

- Really a form of specification intrusion detection
- Based on precise definitions of network protocols
- Can easily detect deviations
- Incorporated into some commercial systems
  - E.g., Snort and Checkpoint

Pluses and Minuses of Specification Detection

+ Allows formalization of what you're looking for
+ Limits where you need to look
+ Can detect unknown attacks
- Only effective when one can specify correct state
- Based on locating right states to examine
- Maybe attackers can do what they want without changing from a "good" state

Customizing and Evolving Intrusion Detection

- A static, globally useful intrusion detection solution is impossible
  - Good behavior on one system is bad behavior on another
  - Behaviors change and new vulnerabilities are discovered
- Intrusion detection systems must change to meet needs

How Do Intrusion Detection Systems Evolve?

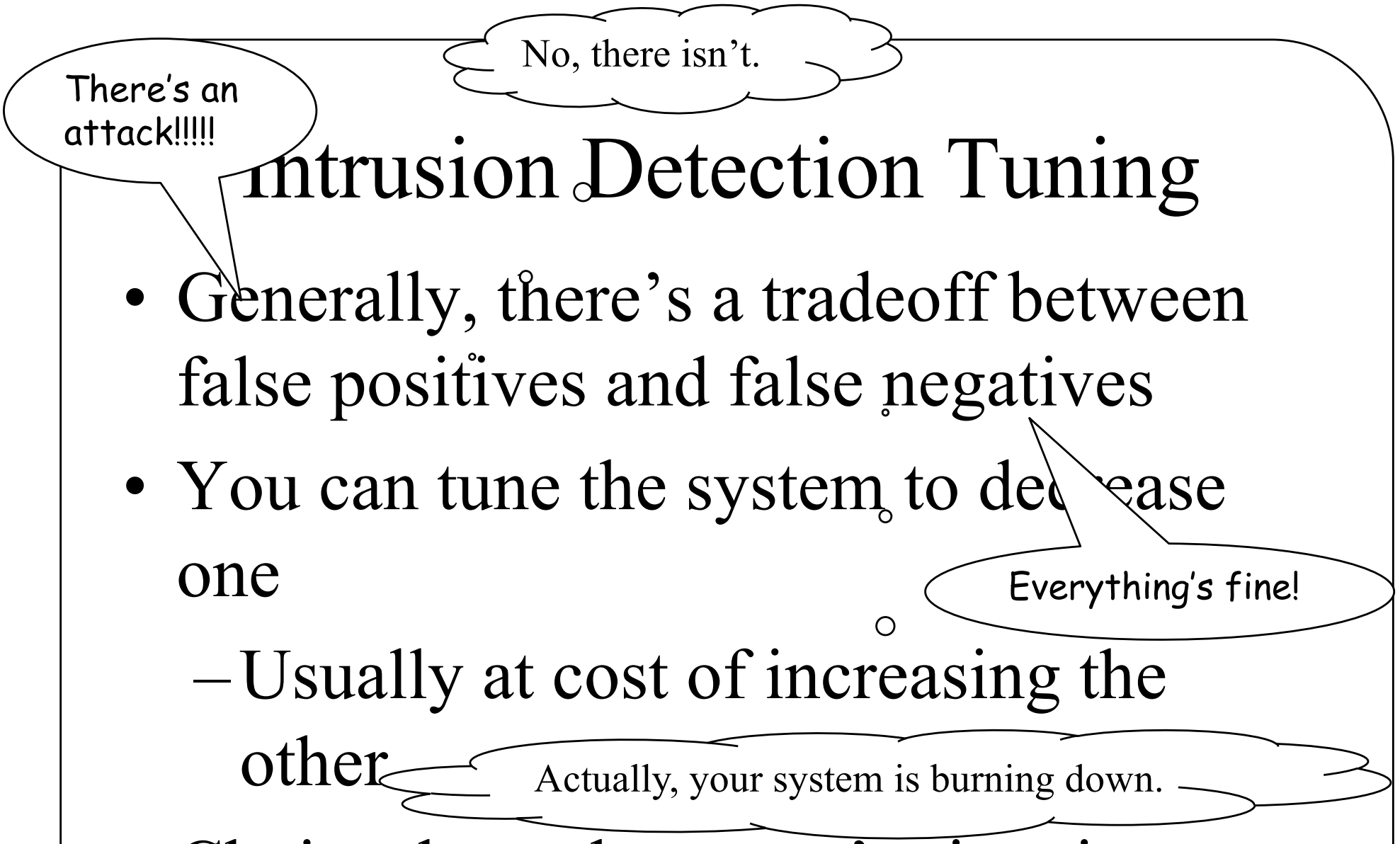
- Manually or semi-automatically
  - New information added that allows them to detect new kinds of attacks
- Automatically
  - Deduce new problems or things to watch for without human intervention

A Problem With Manually Evolving Systems

- System/network administrator action is required for each change
  - To be really effective, not just manual installation
  - More customized to the environment
- Too heavy a burden to change very often
- So they change slowly, akin to software updates

A Problem With Evolving Intrusion Detection Systems

- Very clever intruders can use the evolution against them
- Instead of immediately performing dangerous actions, evolve towards them
- If the intruder is more clever than the system, the system gradually accepts the new behavior
- Possible with manual changing systems, but harder for attackers to succeed


There's an attack!!!!!
No, there isn't.
Intrusion Detection Tuning
• Generally, there's a tradeoff between false positives and false negatives
• You can tune the system to decrease one
– Usually at cost of increasing the other
Everything's fine!
Actually, your system is burning down.
• Choice depends on one's situation

Practicalities of Operation

- Most commercial intrusion detection systems are add-ons
  - They run as normal applications
- They must make use of readily available information
  - Audit logged information
  - Sniffed packets
  - Output of systems calls they make
- And performance is very important

Practicalities of Audit Logs for IDS

- Operating systems only log certain stuff
- They don't necessarily log what an intrusion detection system really needs
- They produce large amounts of data
  - Expensive to process
  - Expensive to store
- If attack was successful, logs may be corrupted

What Does an IDS Do When It Detects an Attack?

- Automated response
  - Shut down the "attacker"
  - Or more carefully protect the attacked service
- Alarms
  - Notify a system administrator
    - Often via special console
  - Who investigates and takes action
- Logging
  - Just keep record for later investigation

Consequences of the Choices

- Automated
  - Too many false positives and your network stops working
  - Is the automated response effective?
- Alarm
  - Too many false positives and your administrator ignores them
  - Is the administrator able to determine what's going on fast enough?
- Logging
  - Doesn't necessarily lead to any action

How Good Does an IDS Have To Be?

- Depends on what you're using it for
- Like biometric authentication, need to trade off false positives/false negatives
- Each positive signal (real or false) should cause something to happen
  - What's the consequence?

False Positives and IDS Systems

- For automated response, what happens?
- Something gets shut off that shouldn't be
  - May be a lot of work to turn it on again
- For manual response, what happens?
- Either a human investigates and dismisses it
- Or nothing happens
- If human looks at it, can take a lot of his time

Consider a Case for Manual Response

- Your web site gets 10 million packets per day
- Your IDS has a FPR of .1% on packets
  - So you get 10,000 false positives/day
- Say each one takes one minute to handle
- That's 166 man hours per day
  - You'll need 20+ full time experts just to weed out false positives

What Are Your Choices?

- Tune to a lower FPR
  - Usually causing more false negatives
  - If too many of those, system is useless
- Have triage system for signals
  - If first step is still human, still expensive
  - Maybe you can automate some of it?
- Ignore your IDS' signals
  - In which case, why bother with it at all?

Intrusion Prevention Systems

- Essentially a buzzword for IDS that takes automatic action when intrusion is detected
- Goal is to quickly take remedial actions to threats
- Since IPSs are automated, false positives could be very, very bad
- "Poor man's" version is IDS controlling a firewall

Sample Intrusion Detection Systems

- Snort
- Bro
- RealSecure ISS

Snort

- Network intrusion detection system
- Public domain
  - Designed for Linux
  - But also runs on Windows and Mac
- Designed for high extensibility
  - Allows easy plug-ins for detection
  - And rule-based description of good & bad traffic
- Very widely used

Bro

- Like Snort, public domain network based IDS
- Developed at LBL
- Includes more sophisticated non-signature methods than Snort
- More general and extensible than Snort
- Maybe not as easy to use

RealSecure ISS

- Commercial IDS
- Bundled into IBM security products
- Distributed client/server architecture
  - Incorporates network and host components
- Other components report to server on dedicated machine

Is Intrusion Detection Useful?

- Widely criticized
- But also best practices usually call for their use
- Signature-based IDS especially criticized
- Thoughtless use brings little benefit
- Requires tuning, updating, and intelligent analysis of IDS outputs to really help

Which Type of Intrusion Detection System Should I Use?

- NIST report[1] recommends using multiple IDSs
  - Preferably multiple types
    - E.g., host and network
- Each will detect different things
  - Using different data and techniques
- Good defense in depth

[1] http://csrc.nist.gov/publications/nistir/nistir-7007.pdf

The Future of Intrusion Detection?

- General concept has never quite lived up to its promise
- Yet alternatives are clearly failing
  - We aren't keeping the bad guys out
- So research and development continues
- And most serious people use them
  - Even if they are imperfect

Conclusions

- Intrusion detection systems are helpful enough that those who care about security should use them
- They are not yet terribly sophisticated
  - Which implies they aren't that effective
- Much research continues to improve them
- Not clear if they'll ever achieve what the original inventors hoped for