# CS530 Intrusion Detection

## **Bill Cheng**

http://merlot.usc.edu/cs530-s10



Copyright © William C. Cheng

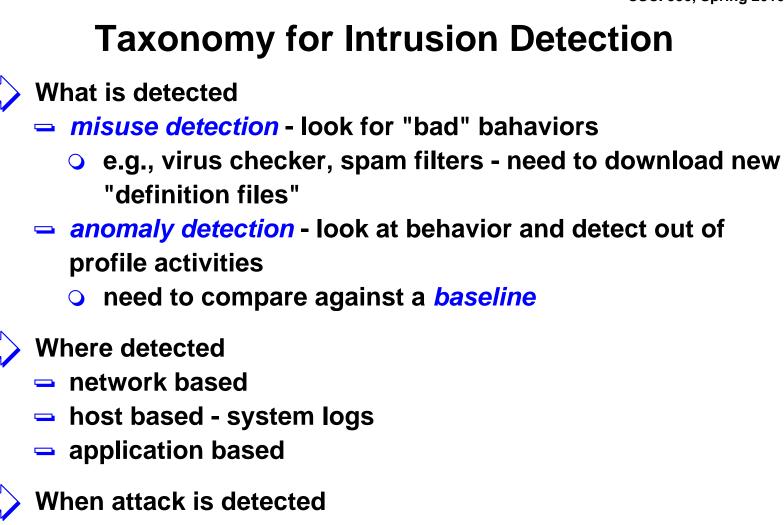
#### **Intrusion Detection**

Security enforcement mechanisms are not foolproof, so we need a way of knowing when they are not working
 or even better, before they stop working

We need ways to detect insider misuse

- detect suspecious activities
  - e.g., is this employee selling information?





- 🛥 real time
- after the fact / post mortem

3 **T T** 

🛏 Copyright © William C. Cheng 🛛

#### **Basis for Detecting Attack**

- Systems operating normally
  - activity conforms to statistically predictable patterns
  - actions do not include attempts to subvert policy
  - actions of processes conform to the policies regarding what they are allowed to do
  - e.g., when system is under attack, will see unusual amount of denied accesses

## **Rating ID Systems**

#### False positives

- normal activity flagged as intrusion
  - affects adminstrator workload
    - e.g., port scanners if you don't have the vulnerability, do not raise alarm
- e.g., spam filtering
  - I filter out all HTML-only e-mails
- too many of these denial of service on yourself
- "the boy who cried wolf"

#### False negatives

attacks that are not detected



## **Anomaly Detection**

How it works

- analyze baseline characteristics of system or user behavior and record
  - o need to have an abstraction or a model
- compare current characteristics and behavior against baseline and determine if it's within tolerance
  - or is it just a statistical fluctuation
- flag differences
- Why it is hard
  - deciding how to characterize behavior so that changes reflect intrusions and not normal changes in activities
- Credit card companies do this all the time



## **Metrics**

- Threshold metrics
  - number of failed access attempts
    - e.g., confiscate ATM card after 3 bad PINs
  - bandwidth consumed
    - e.g., can be used to detect misuses from within

State change probabilities (Markov models)

- requires training by analyzing normal traces (system logs)
  - there are systems that can be trained while monitoring
- Iooking for transitions that don't seem to follow the normal pattern





#### **Misuse Detection**

- Whether activities or code is violate site policy
  - rule based
    - e.g., if A is followed by B and if B is followed by C, flag it
  - signature based
- Problems
  - can only detect attacks known in advance
  - virus checkers are usually signature based
    - can protect against write to boot sector
  - many more false negatives (subject to definition)
    - vendor's definition?
- Strengths
  - tend to have fewer false positives



## **Collecting Input Data**

- > Audit vs. Intrusion Detection
- > Network based ID
- Host based ID
- **Application based ID**



#### **Network Based ID**

Often based on network sniffing

#### listening to network traffic as it goes by a sensor node

- could be placed in routers or other network components
- e.g., SNORT packet sniffer

#### 🛥 issues

- o placement
  - be careful with switched Ethernet
  - wireless channel can be asymmetric
- Ioad
  - may log only summary information to reduce load
    e.g., IP traceback
- encrypted traffic (such as IPSec)
- (cont...)



🛏 Copyright © William C. Cheng 🛛

## Network Based ID (Cont...)

- issues (cont...)
  - O determining intent
    - e.g., if a message to port 24 (SMTP) does not look like e-mail, flag it
    - e.g., in HTTP, turn on encryption (but don't really encrypt) - ID will ignore these messages!
       can use this "feature" for tunneling



#### Host Based ID

- > We have better understanding of these
  - because hosts are usually not an open system (unlike networks)
  - but break-ins can be covered up easier (unlike networks)
- Scan system and application logs
- Report on system state
  - e.g., load, who are logged in
- Report activity to ID system
- Issues
  - only get what applications already put into logs
  - might not understand the intent of an action



## **Application Based ID**

Application determines what to report to ID system

based on a policy

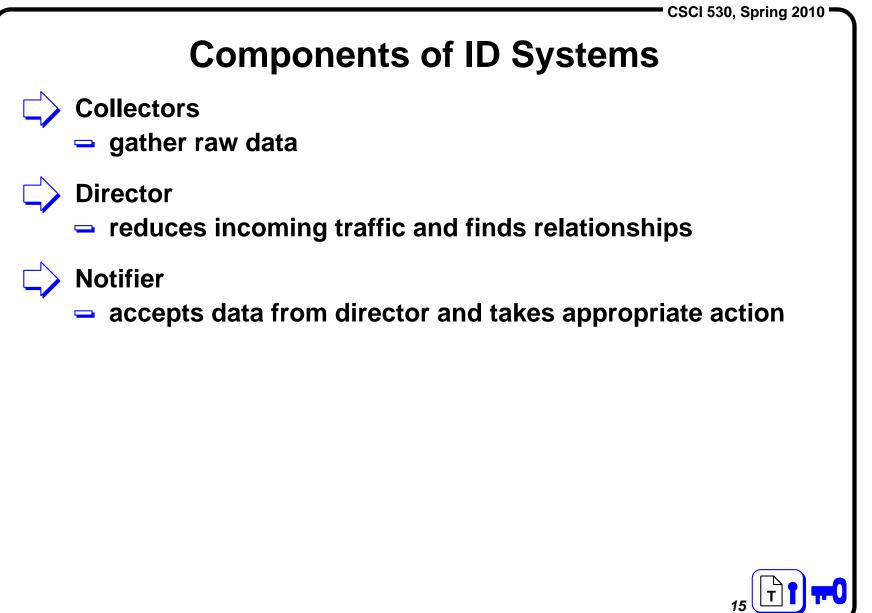
- **Drawbacks** 
  - equires application involvement (some applications will not report)
    - authorization functions like GAA-API can help address this limitation
  - Benefits
    - application understands the objects and entities to which policies apply



#### **Issues In Intrusion Detection**

- Collecting data on and reporting events
  - interoperability issues
  - languages, e.g. CIDF
- Reducing data
  - to reduce network traffic consumed
    - o consider overhead
  - 🛥 summarize data
    - e.g., 10 of the following messages have been seen
    - o finding relationships
  - what have you filtered out that shouldn't be filtered out?





## **Advanced IDS Models**

**Distributed detection** 

- combining host and network monitoring (DIDS)
- autonomous agents (Crosbie and Spafford)
- COSSACK project at USC/ISI professor Papadopoulos



#### **Intrusion Response**

Intrusion prevention

- it's a marketing buzzword
- > Intrusion response
  - how to react when an intrusion is detected (or an attempt of intrusion)



#### **Possible Responses**

- Notify administrator
- System or network lockdown
  - change firewall rules
- Place attacker in controlled environment
  - quarantine
    - done with worms no outgoing traffic from this node
    - use a Honeypot to attract unsuspecting attacker

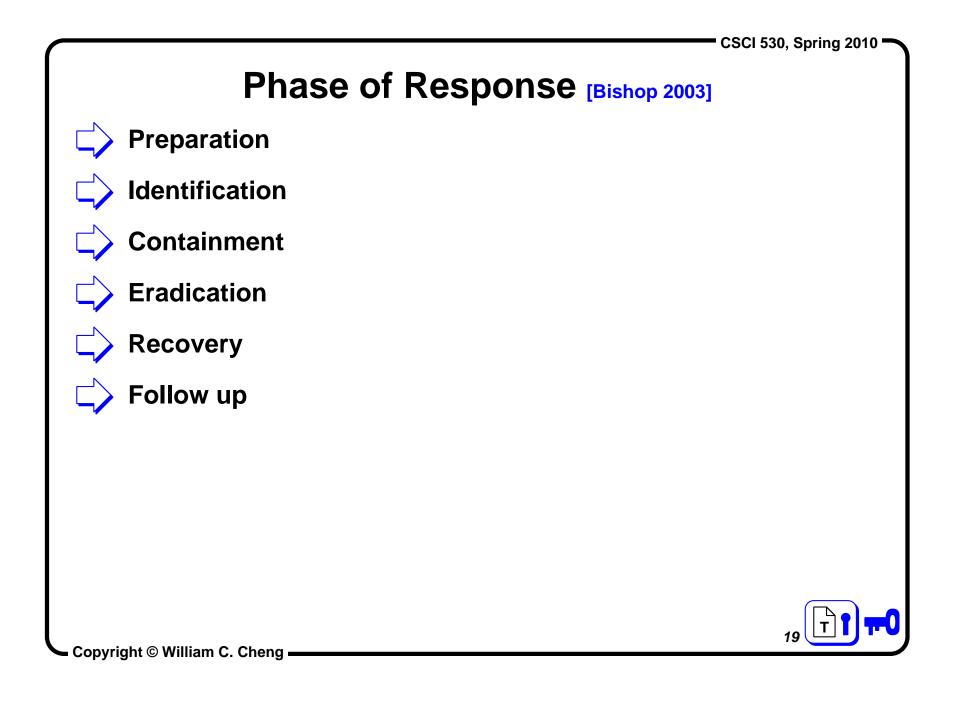
Slow the system for offending processes

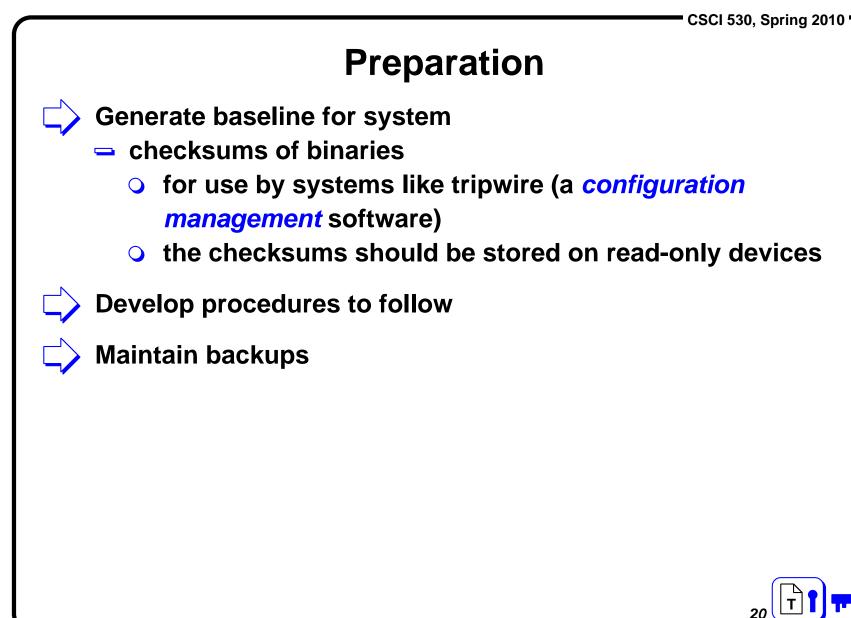
commonly used for SMTP servers - if spam is detected, slow down the connection

Kill the process

often it is more desirable to suspend the process so you can examine memory

🛏 Copyright © William C. Cheng 🛛





#### Identification

- > This is the role of the ID system
  - detect attack
  - characterize attack
  - try to assess motives of attack
    - e.g., making your system a zombie vs. identity theft
    - isolate and observe
      - can use a Honey Pot
      - may have liability issues
  - determine what has been affected
    - o be careful with the Electronic Privacy Act
      - o you need a warrant to run a Honey Pot?



## Containment

- Passive monitoring
  - to learn intent of attacker
  - learn new attack modes so one can defend against them later
- Constraining access
  - locking down system
  - closing connections (in-bound or out-bound)
  - blocking at firewall, or closer to source (for DDoS attacks)
    - active network (network management application)

#### Combination

 constrain activities, but don't let attacker know that one is doing so (Honeypots, Jail)



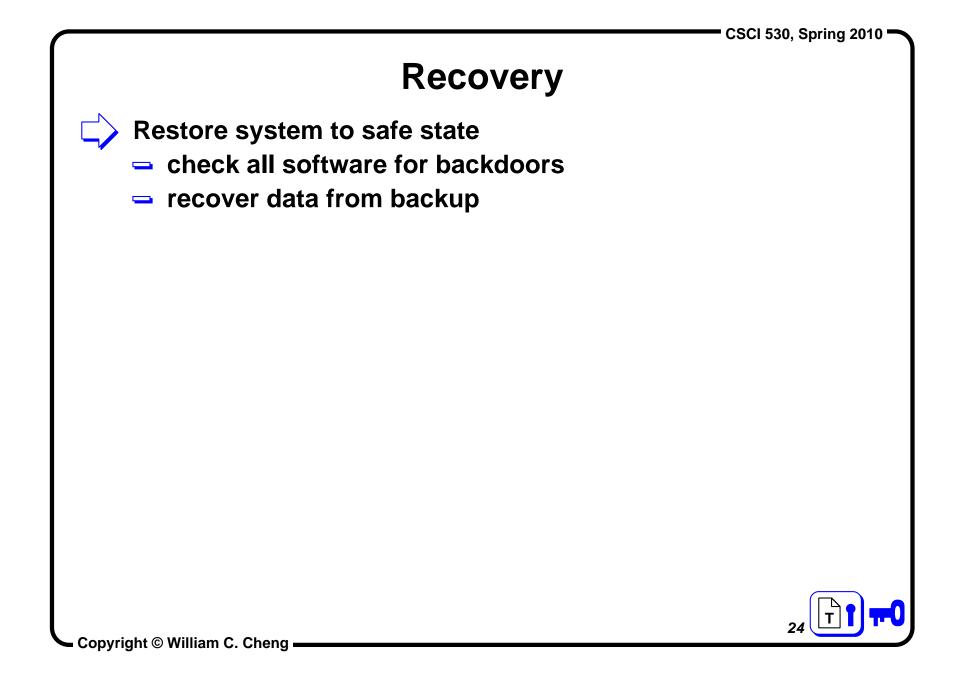
Copyright © William C. Cheng

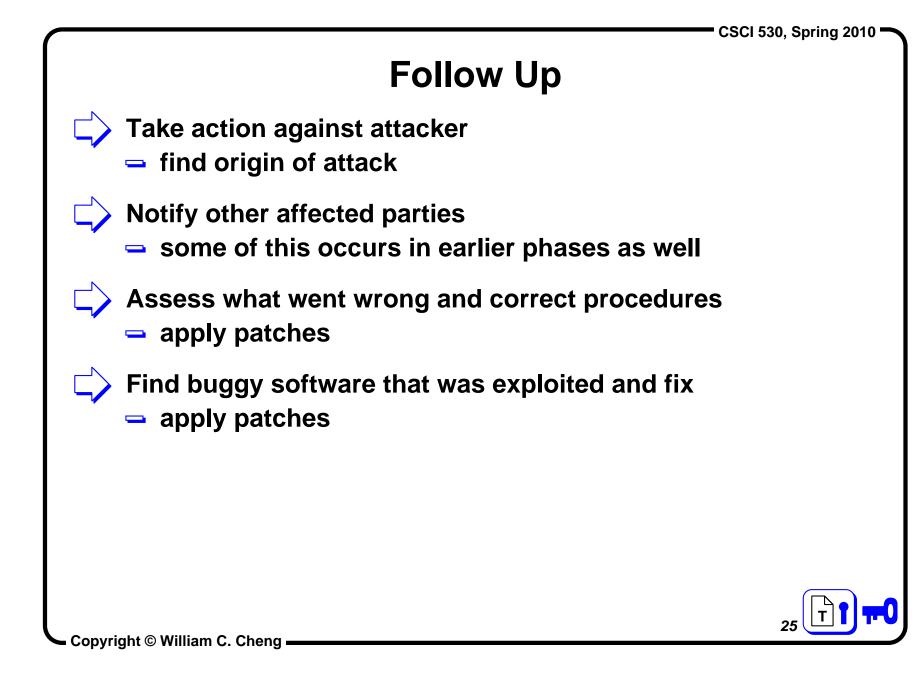


#### **Eradication**

Prevent attack or effects of attack from reoccuring

- Iocking down system (also in containment phase)
- blocking connections at firewall
- isolate potential targets (inverted quarantine)





## **Security for USC/ISI**

- Academic environment
  - open environment
    - people want to run own servers
      - different for departments vs. students
- what protection does your environment need?
  - for inexperienced people, put them behind firewall
- sensitivity of information to be protected
  - student records
  - o medical records (medical school, HIPAA requirements)
- data in student's directories
  - cannot have control over these (unlike for employees)

