Department of
Computer Science

Active System Management

William A. Arbaugh

Aram Khalili

Pete Keleher

Leana Golubchik

Department of Computer Science

Virgil Gligor

Bob Fourney

Department of Electrical and Computer Engineering

University of Maryland, College Park

Talk Overview

- Measuring Security Vulnerabilities
  - Robert Fourney and Virgil Gligor
- Predicting the Severity of Intrusion Series
  - Hilary Browne and William Arbaugh[1]
- Determining the State of an Information System
- Goals of Active System Management
- Status and Future Work

[1] Joint work with John McHugh and Bill Fithen of CERT/CC

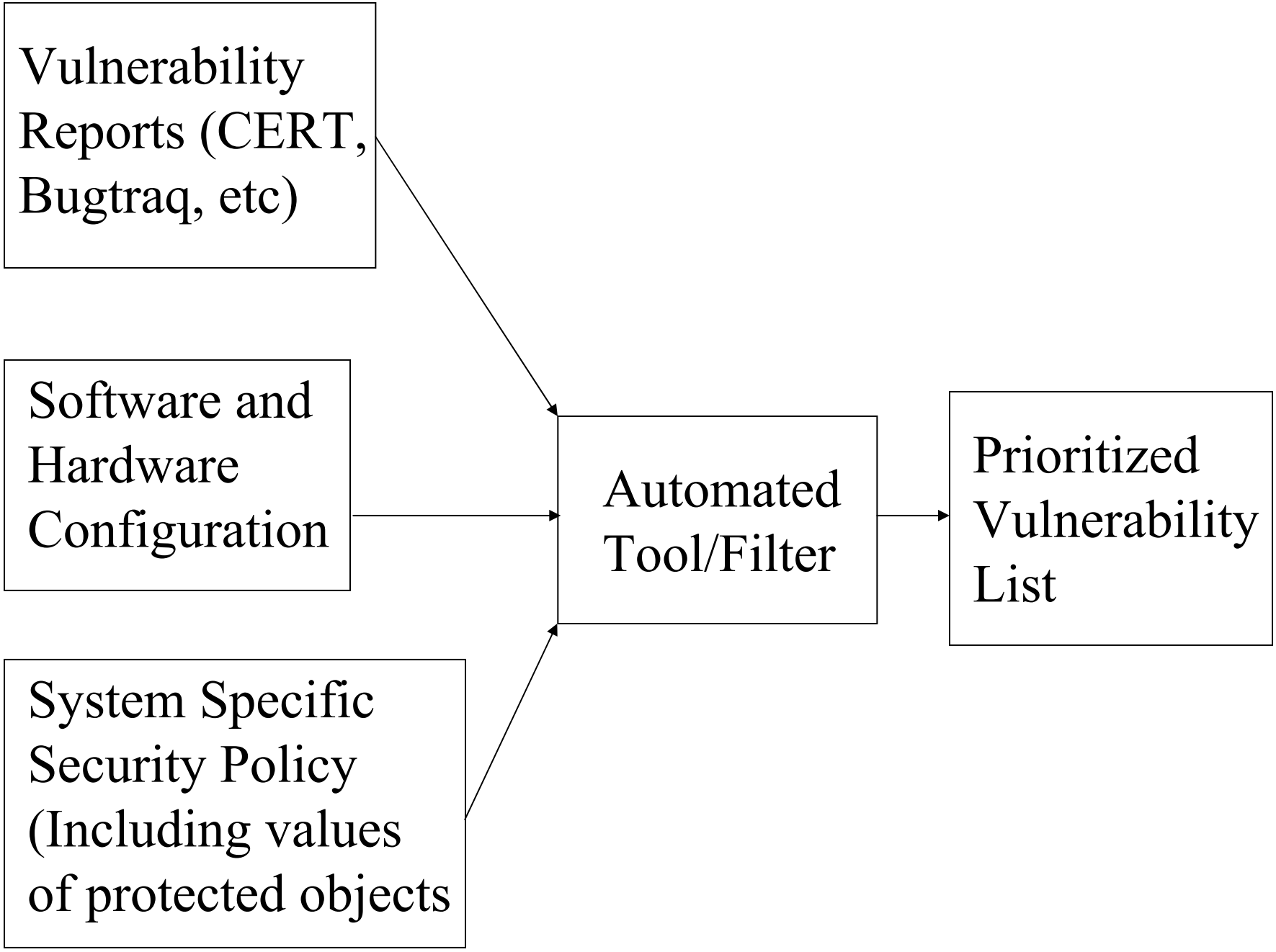
Measuring Security Vulnerabilities

Problem

- The majority of system intrusions are due to "known and patchable vulnerabilities" [Arbaugh *et al*]
- The average computer user is becoming less "computer savvy" [Mehta and Sollins]

Ideal (long term) Solution

- An automated method or tool to aid the local system administrator in prioritizing vulnerabilities, deciding which vulnerabilities to patch, and deciding in what order they should be patched.


Vulnerability Reports (CERT, Bugtraq, etc)
Software and Hardware Configuration
System Specific Security Policy (Including values of protected objects
Automated Tool/Filter
Prioritized Vulnerability List

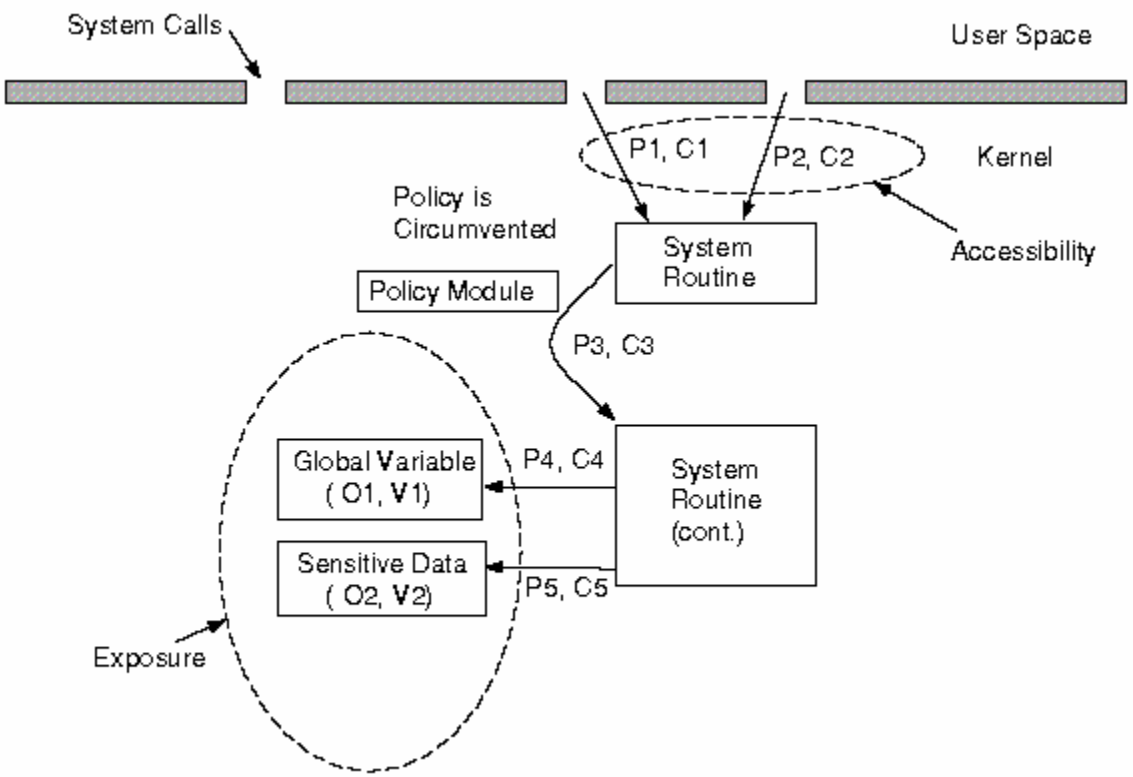
Intermediate Goal

- A method of measuring flaws which enables their effects to be assessed and compared.

Exposure Metric

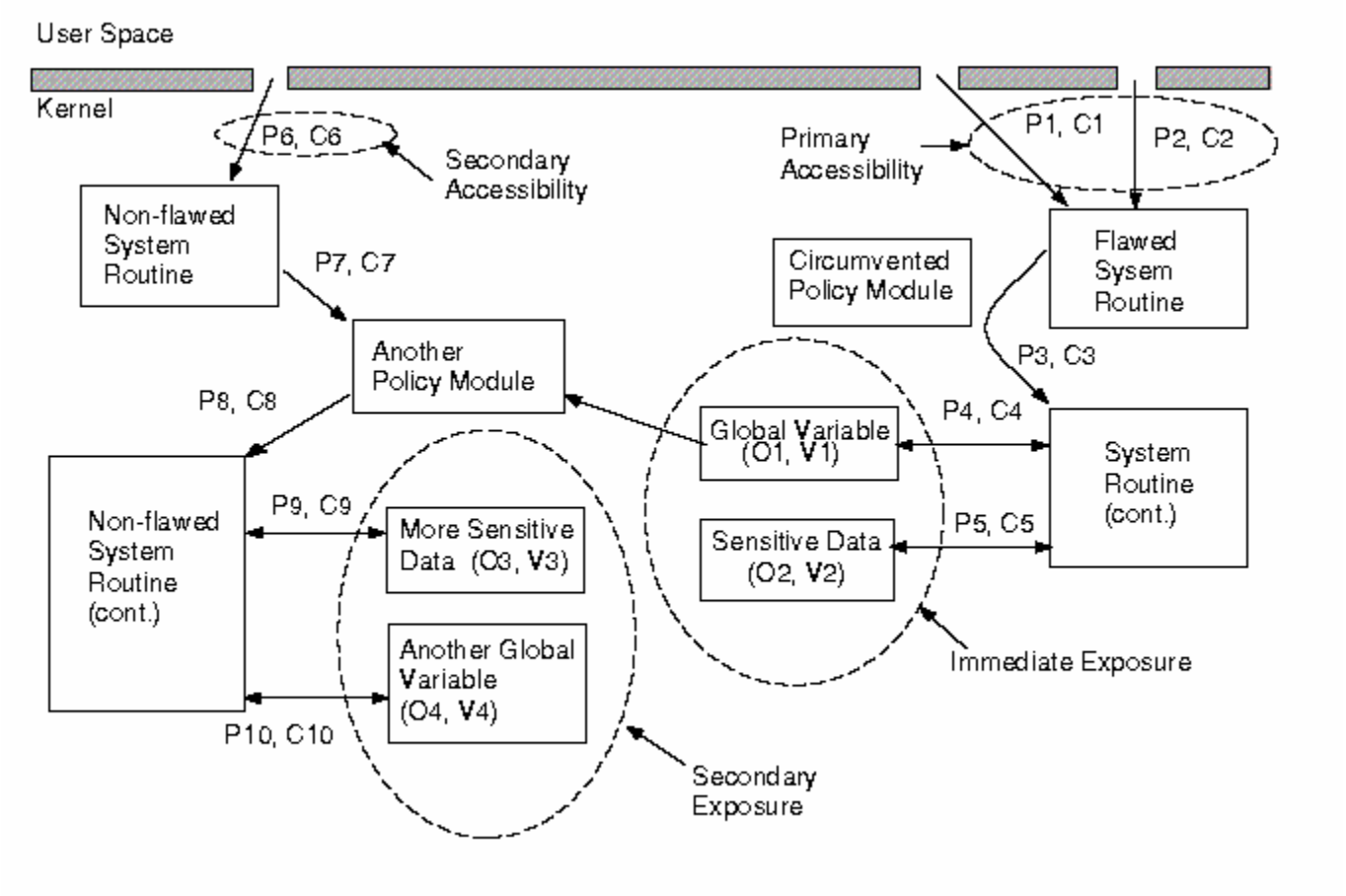
- Measures flaw independently of a formal specification or criteria.
- Measures flaw based on source code analysis.

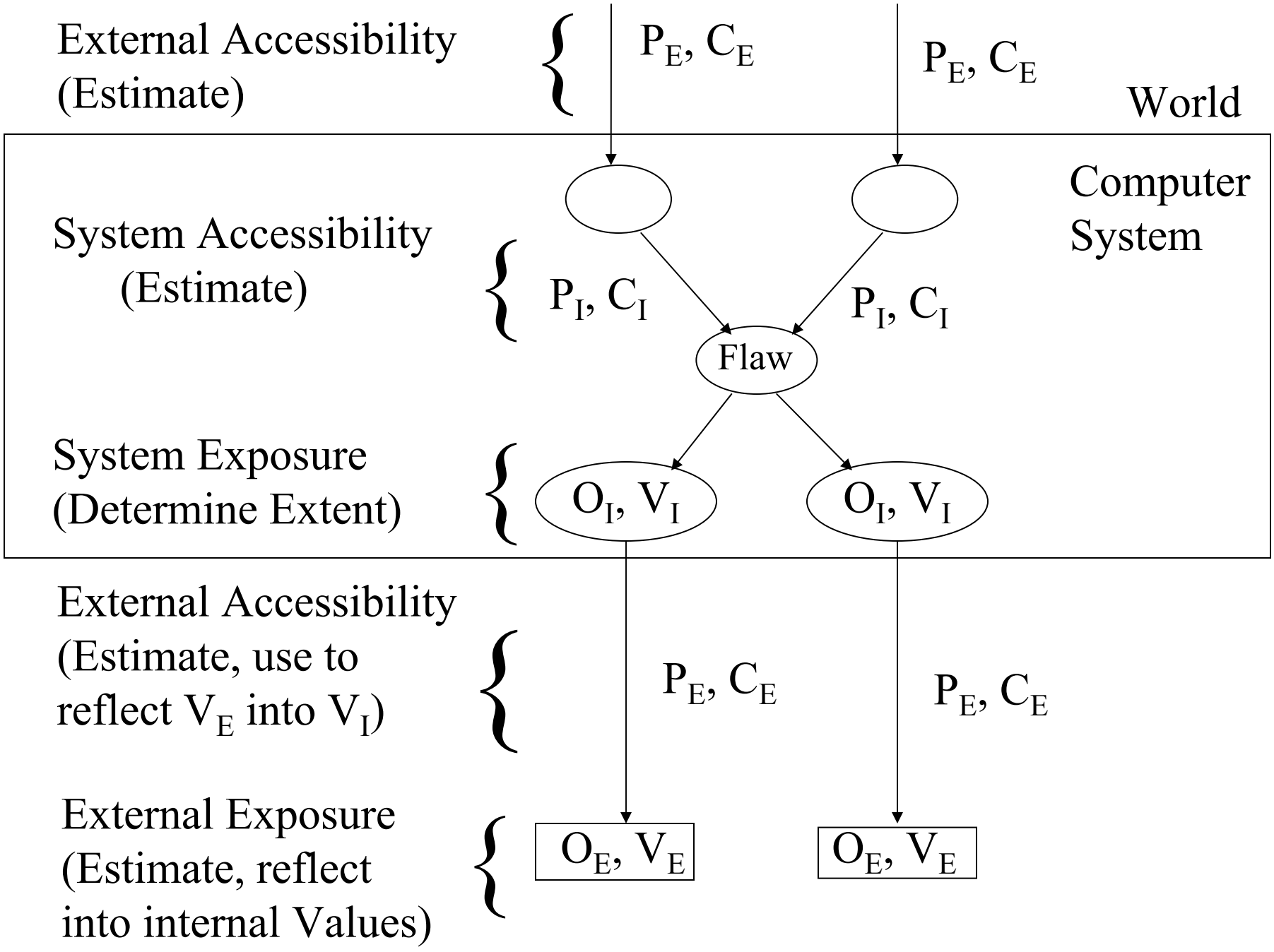
Flawed System Call*



*Also Applies to Application Call

Secondary Exposure

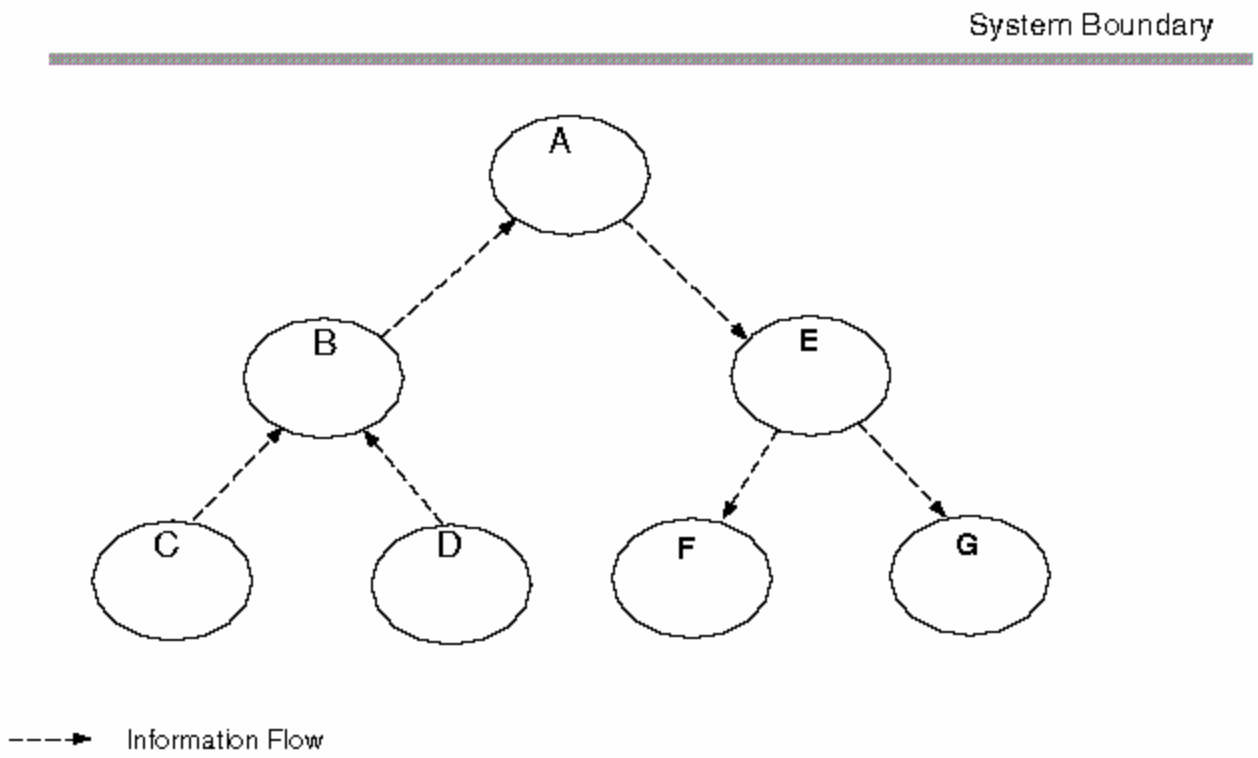



External Accessibility
(Estimate)
P_E, C_E
P_E, C_E
World
Computer
System
System Accessibility
(Estimate)
P_I, C_I
P_I, C_I
Flaw
System Exposure
(Determine Extent)
O_I, V_I
O_I, V_I
External Accessibility
(Estimate, use to
reflect V_E into V_I)
P_E, C_E
P_E, C_E
External Exposure
(Estimate, reflect
into internal Values)
O_E, V_E
O_E, V_E

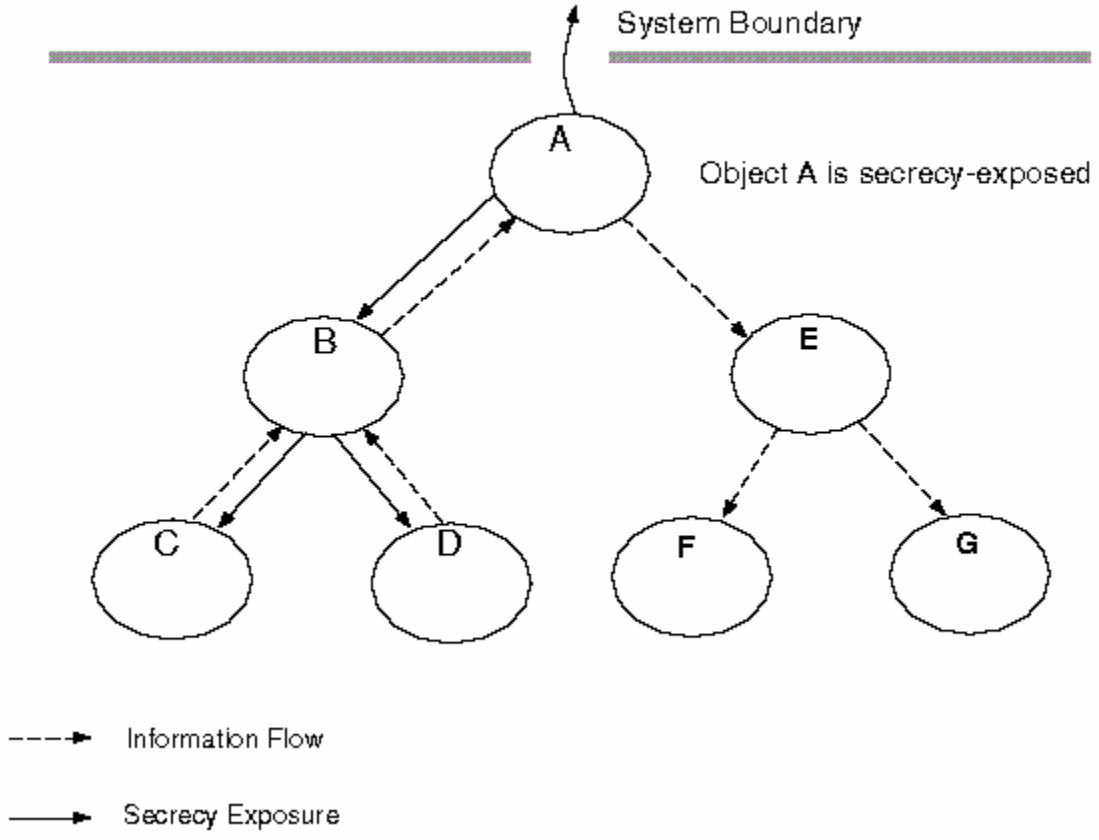
Building Blocks Used

- Information Flow-- occurs whenever the value of an object is obtained, either directly or indirectly, from another object. [Denning]
- Control Flow-- refers to the way in which control is transferred between individual statements and functions within a program [Gupta]
- Functional Dependency-- exists between two functional components, A and B, if the correct implementation (function) of A relies on the correct implementation (function) of B [Parnas]

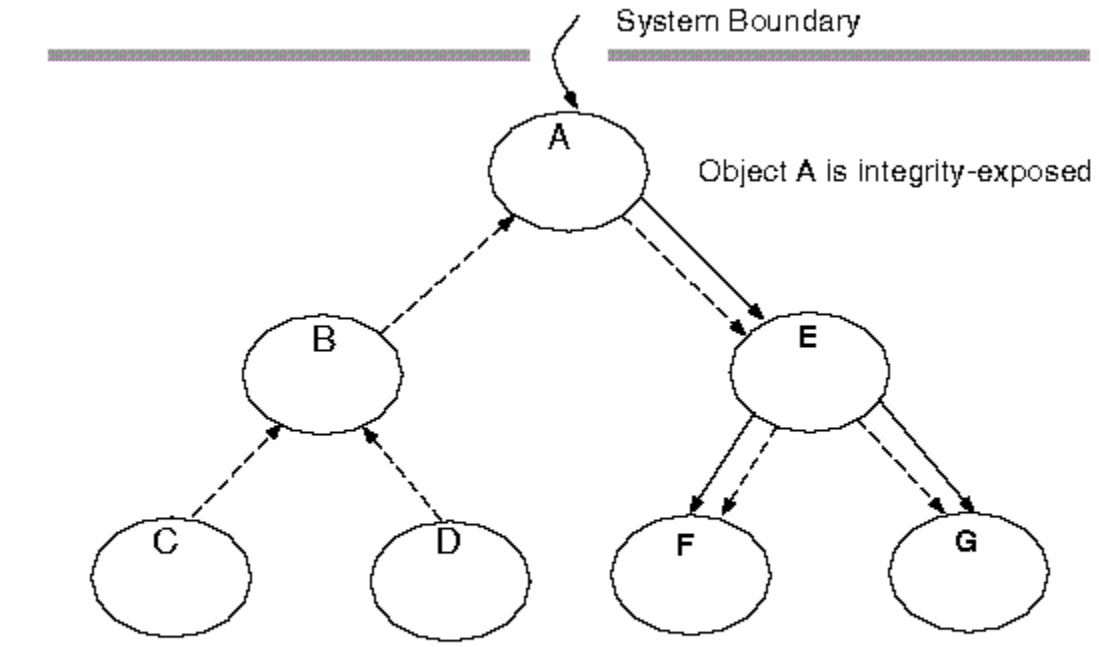
Information Flows Within a System



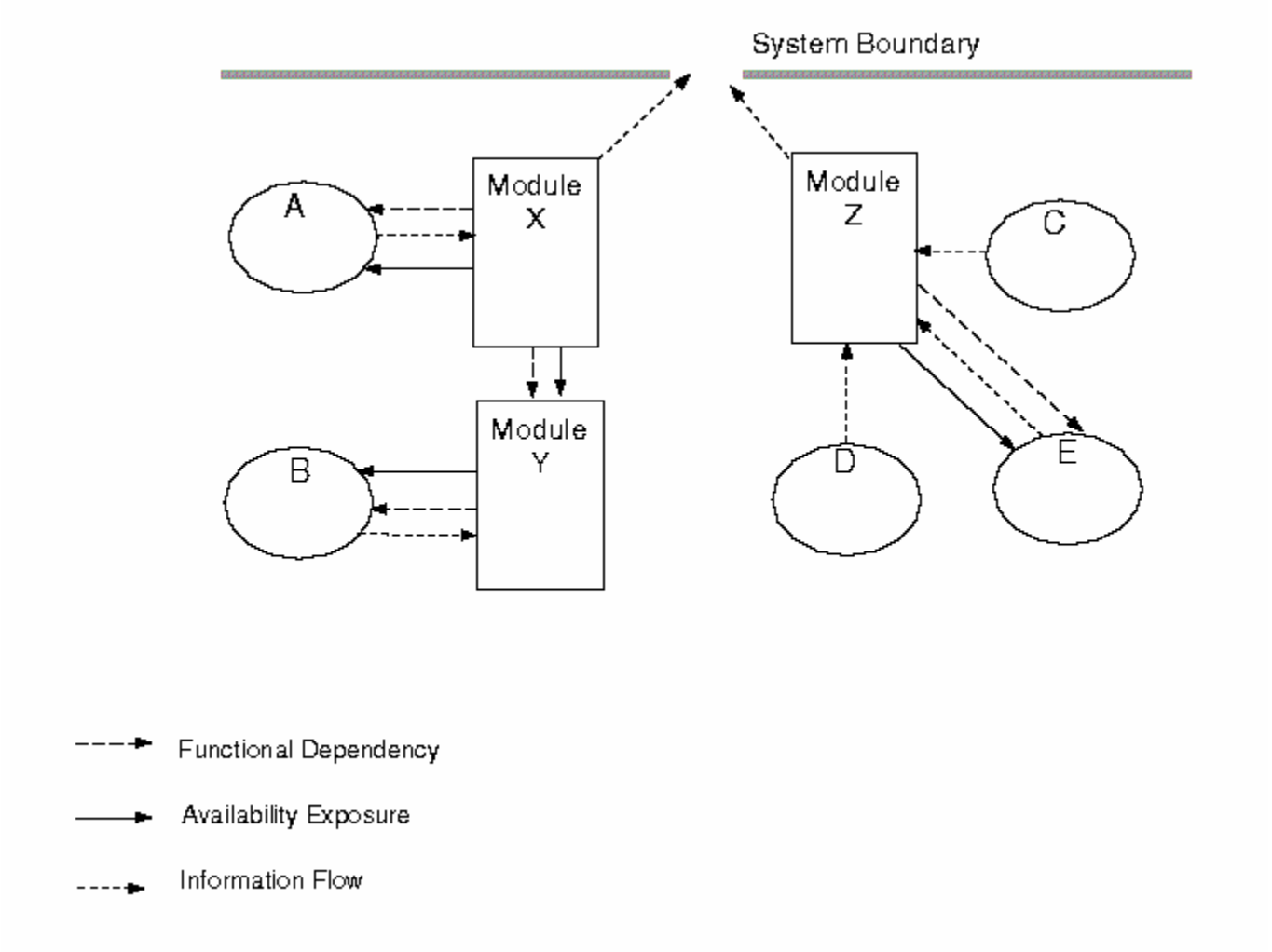
Secrecy-Exposure



Integrity-Exposure



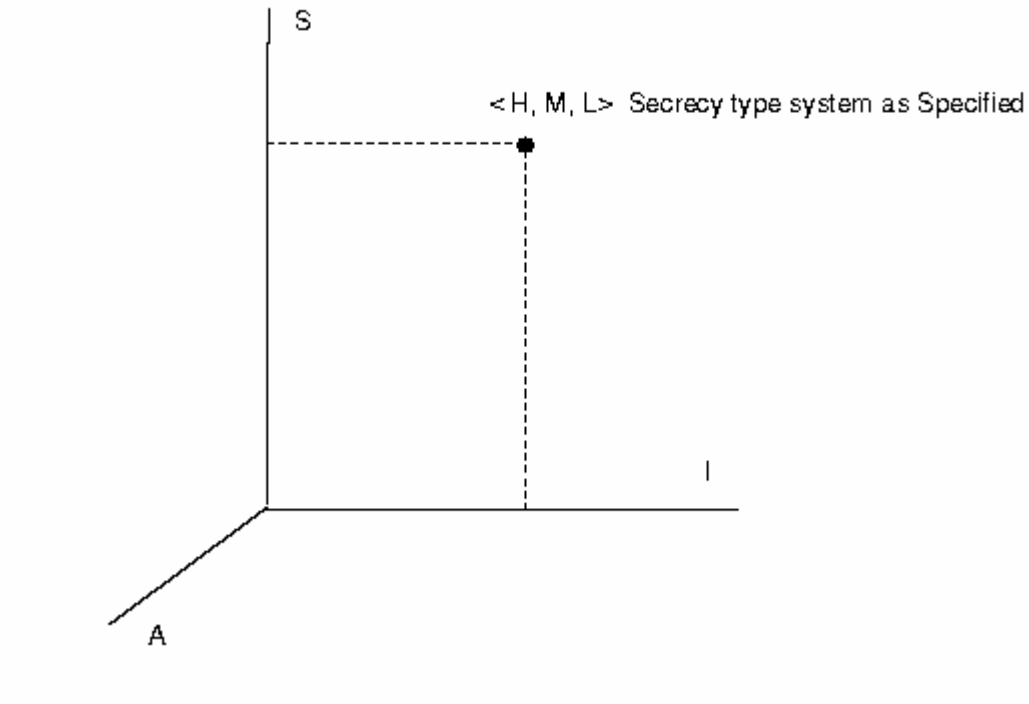
Availability-Exposure



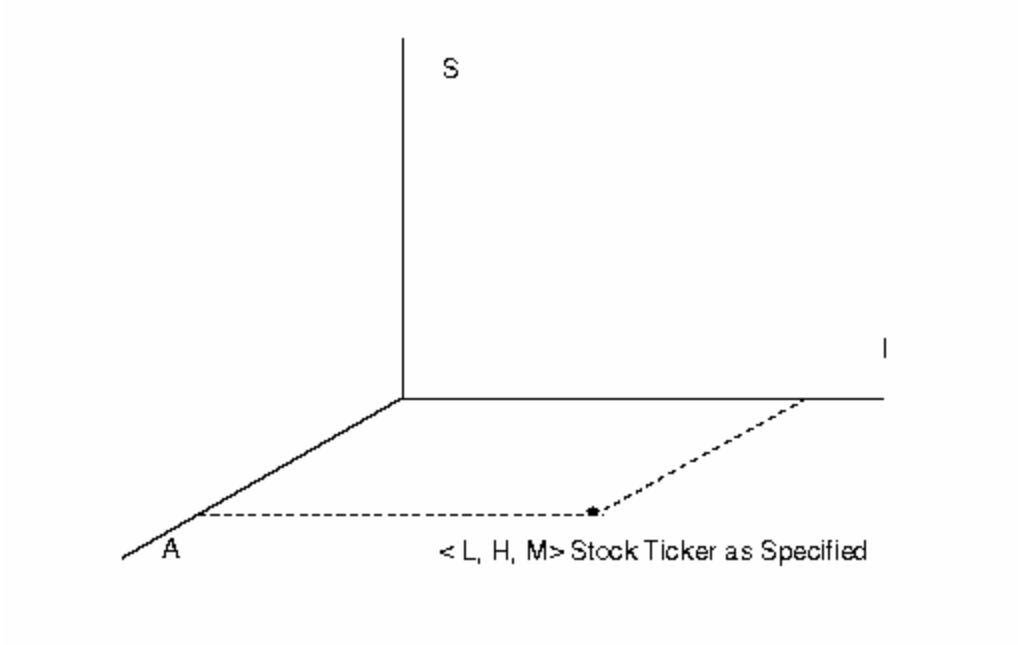
Region of Vulnerability Metric

- Measures effect of flaw relative to specified security level

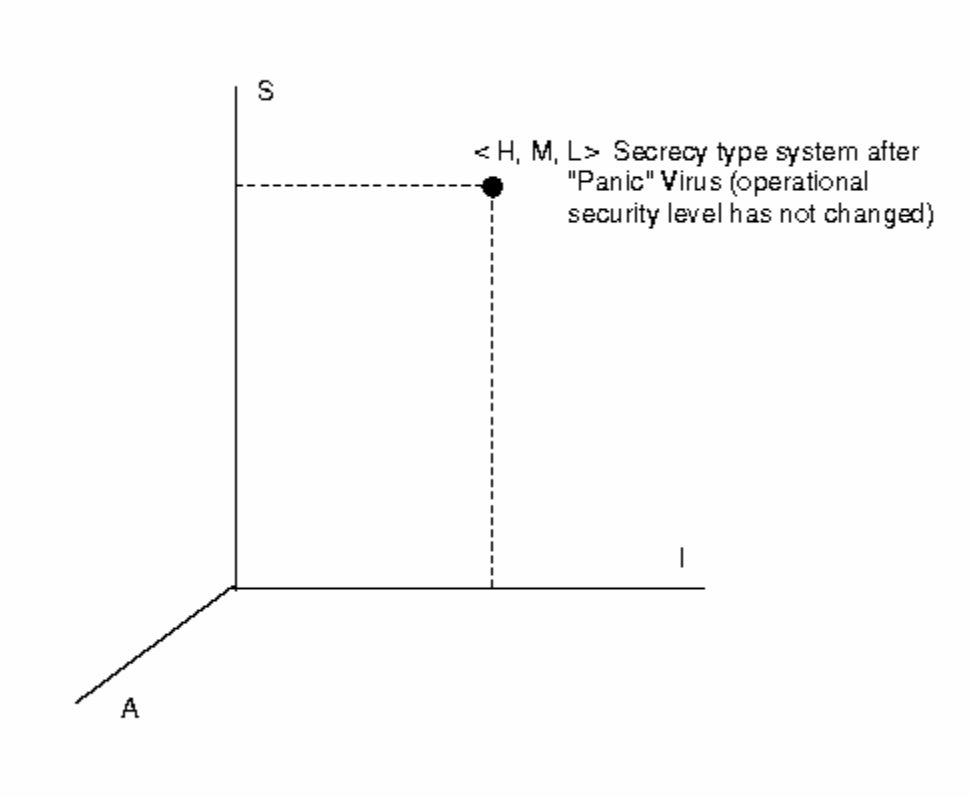
Secrecy type system specification



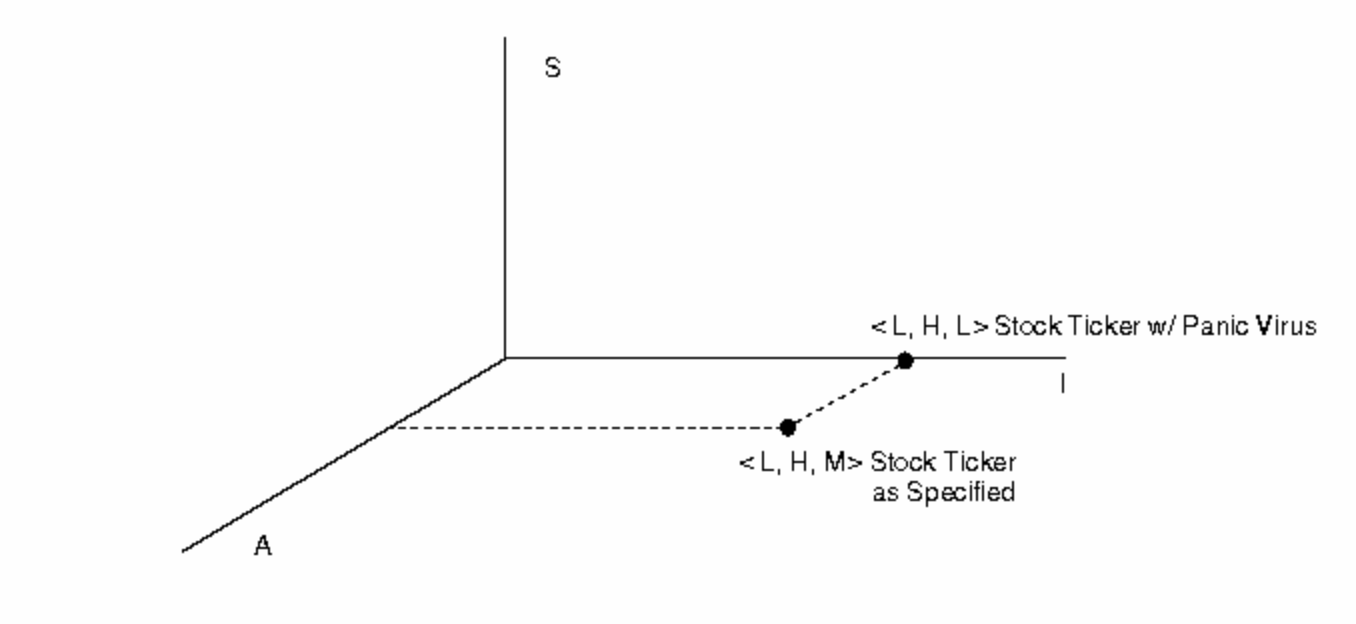
Stock Ticker System Specification



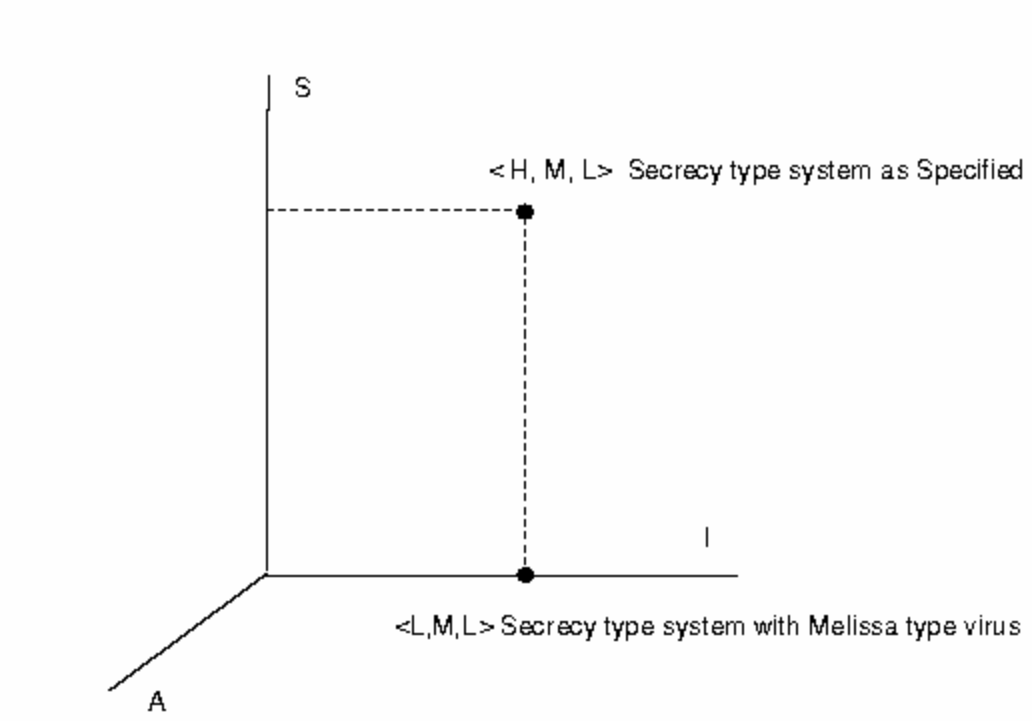
Secrecy System with Panic



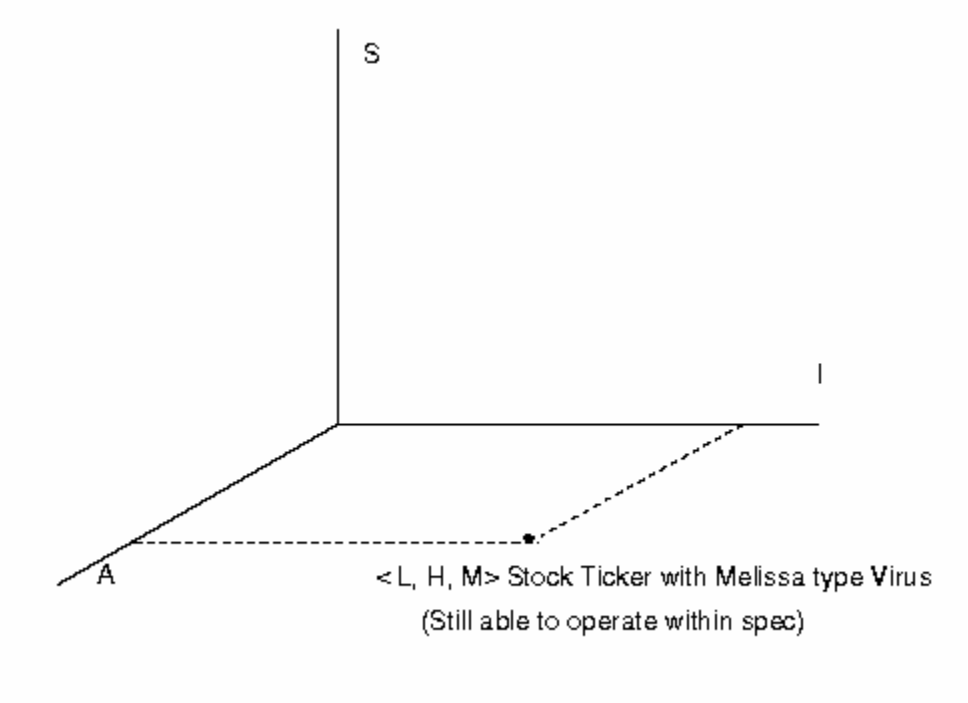
Stock Ticker with Panic



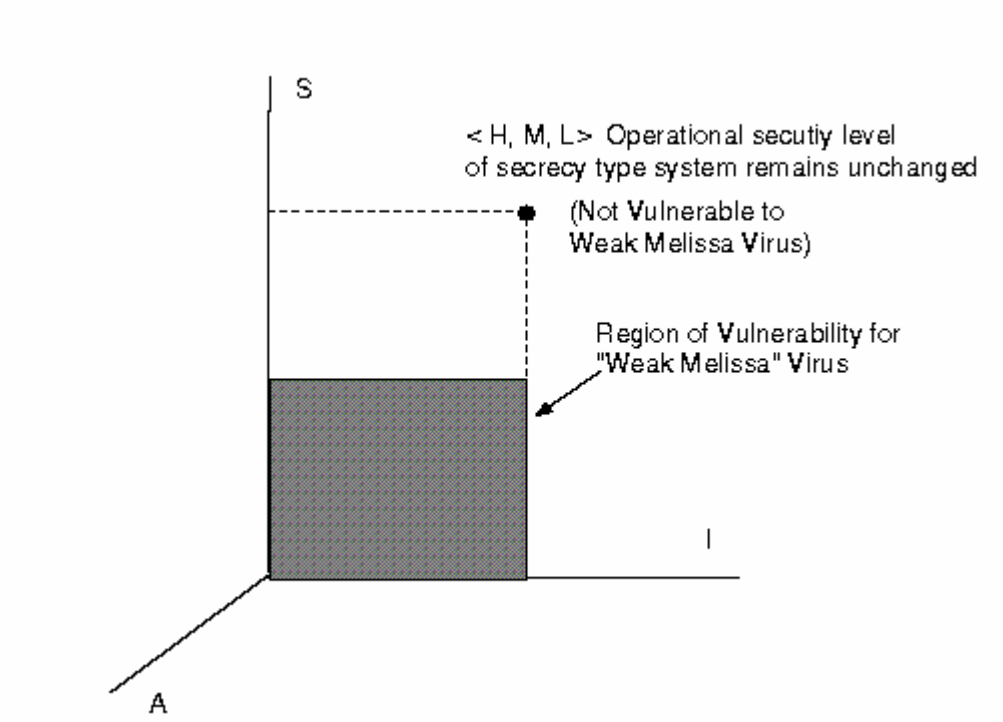
Secrecy with Melissa



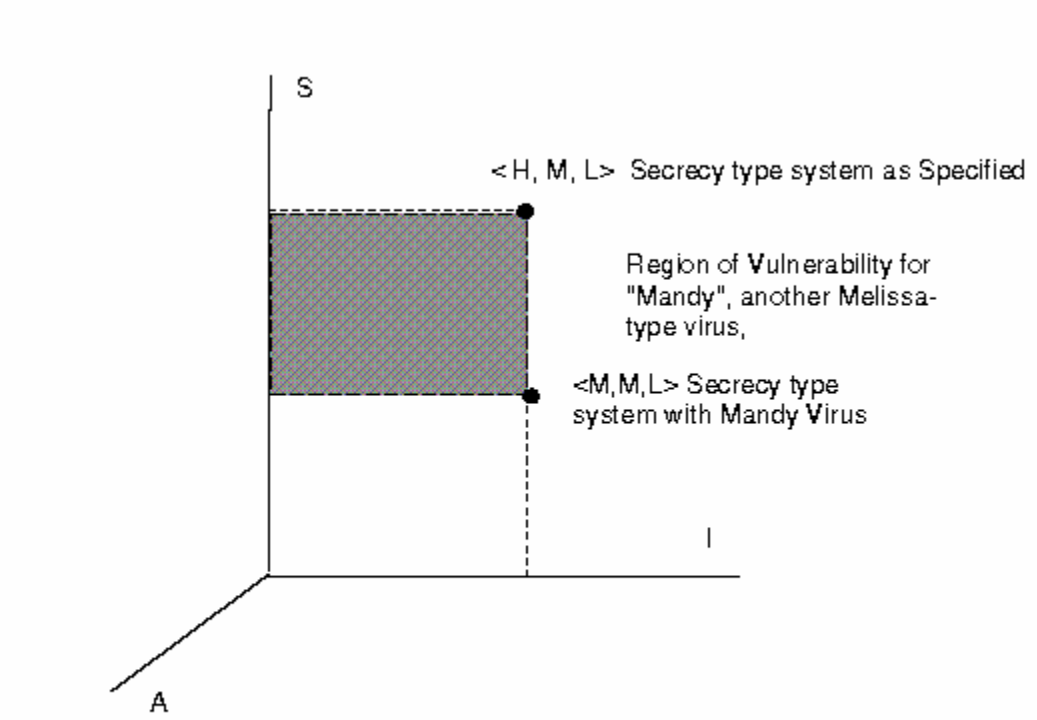
Stock Ticker with Melissa



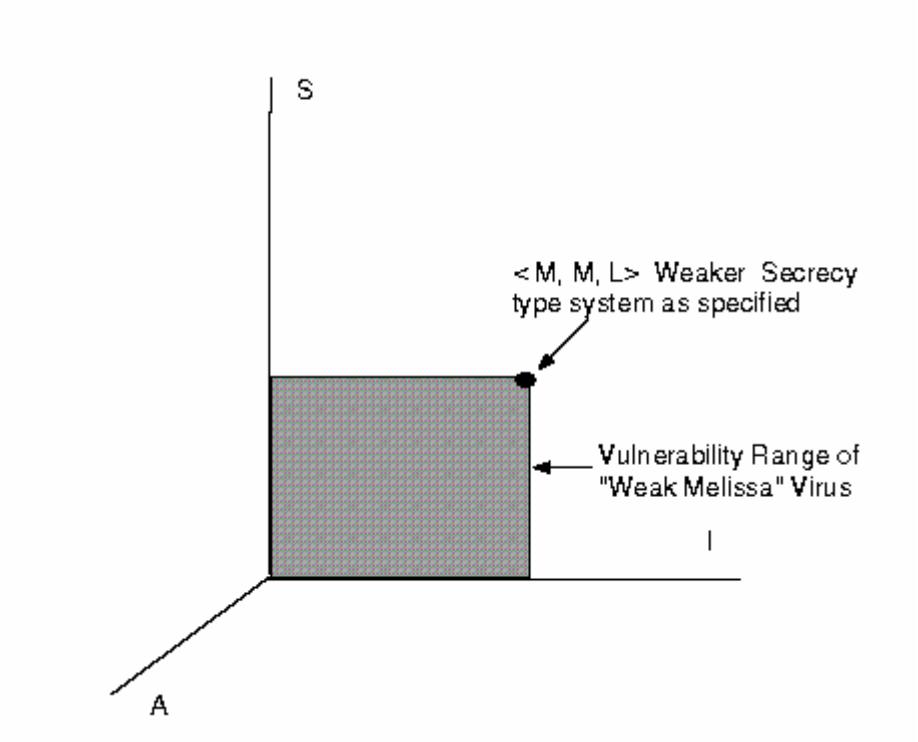
Secrecy System Immune to "Weak Melissa"



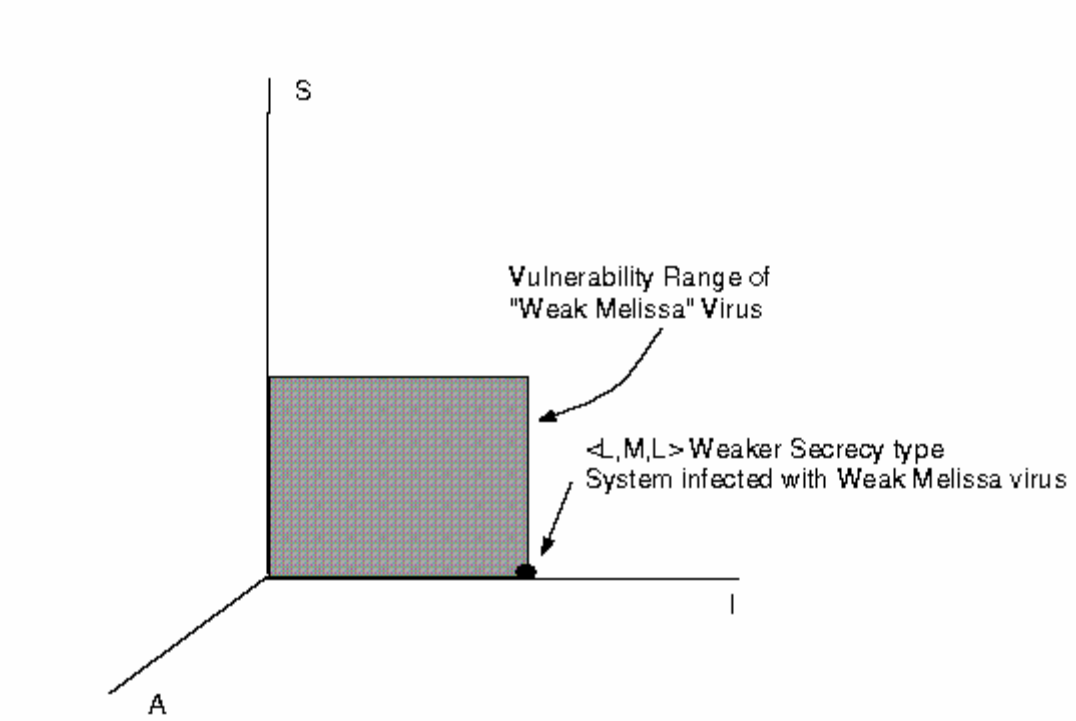
Secrecy System with Mandy



Weaker Secrecy System

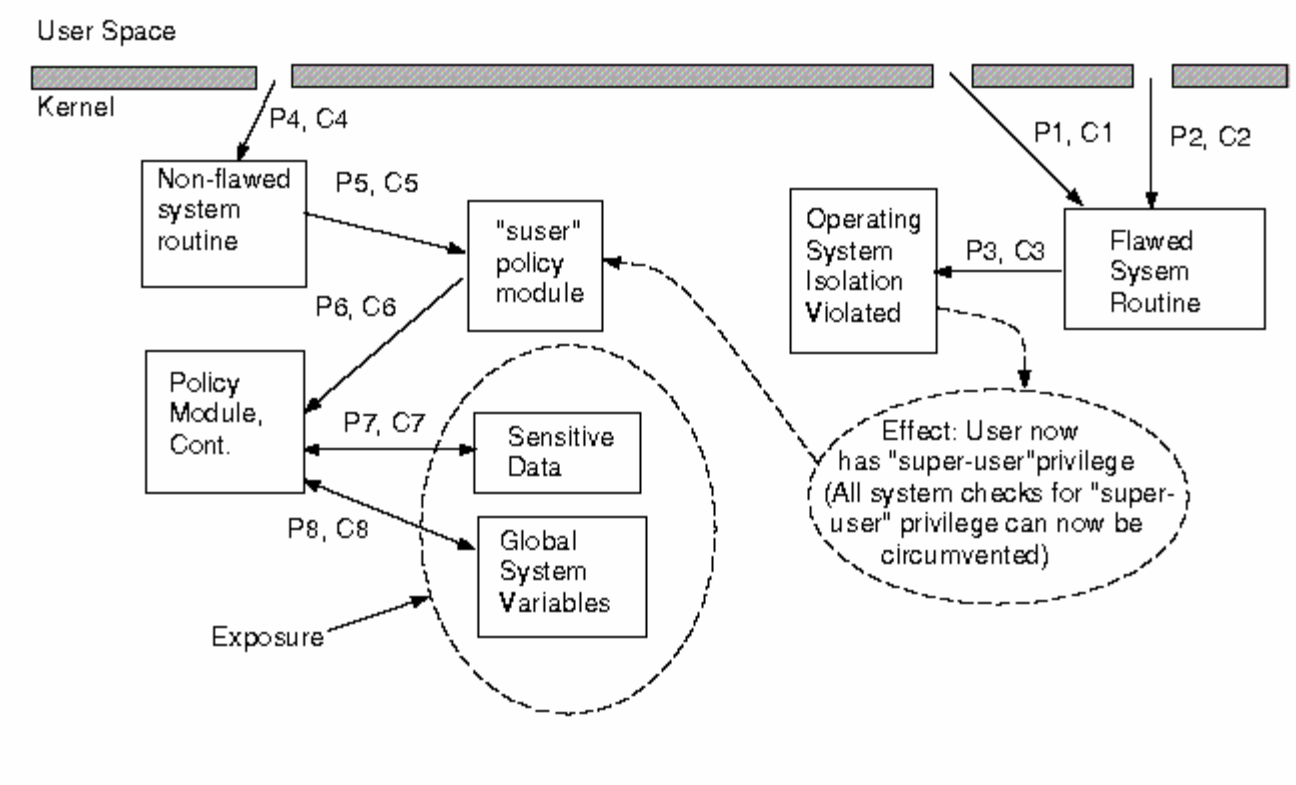


Weaker Secrecy System **not** Immune to “Weak Melissa”

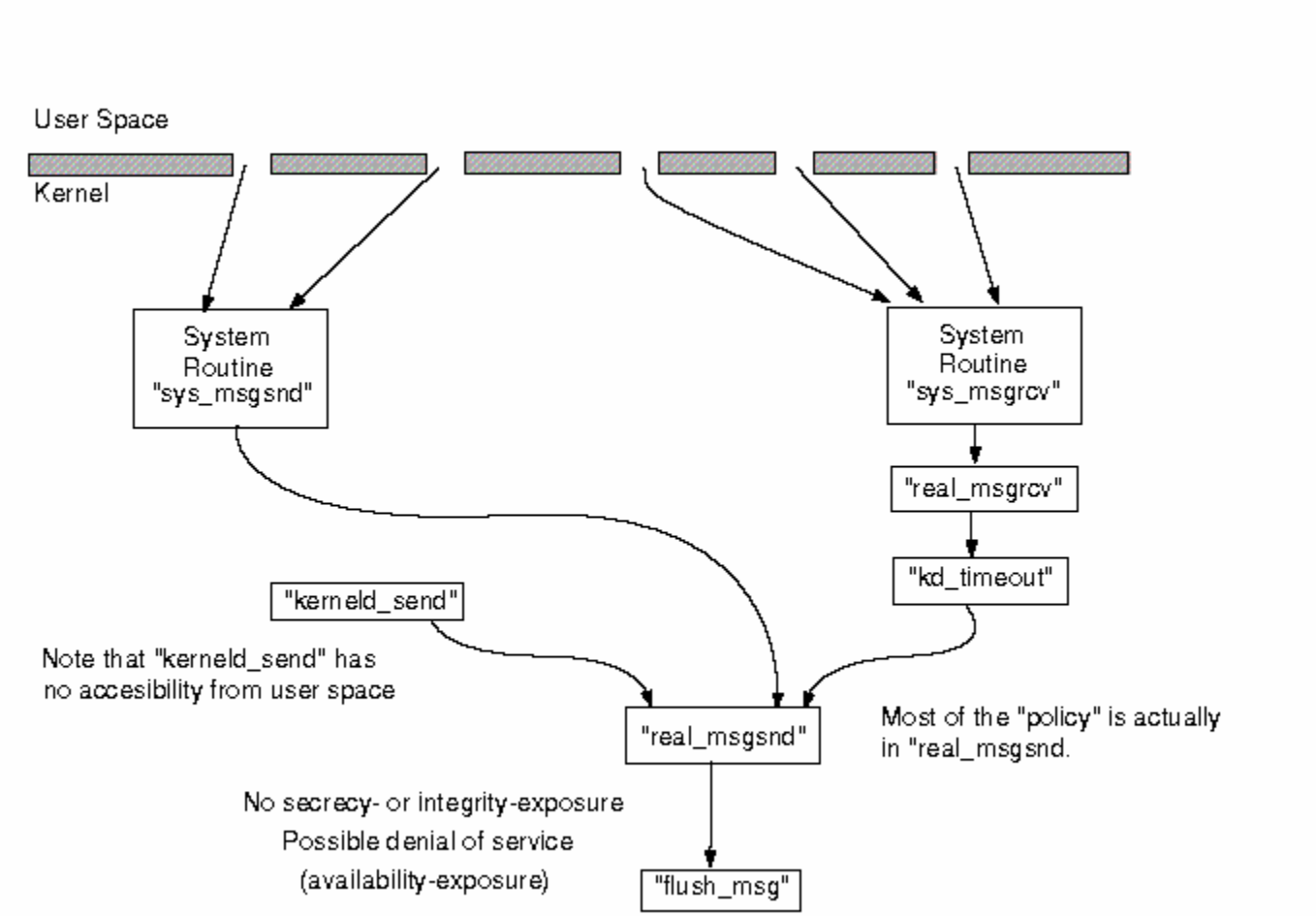


Examples

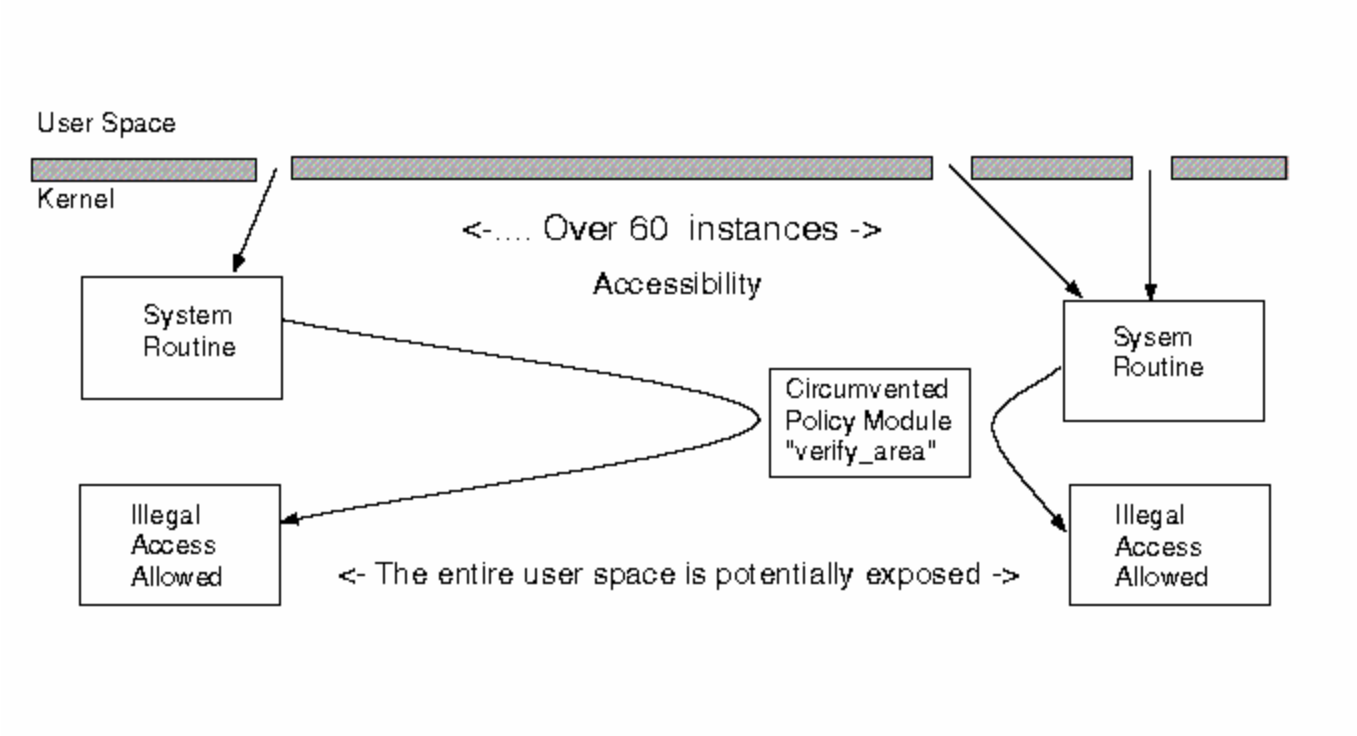
System Isolation Exposure



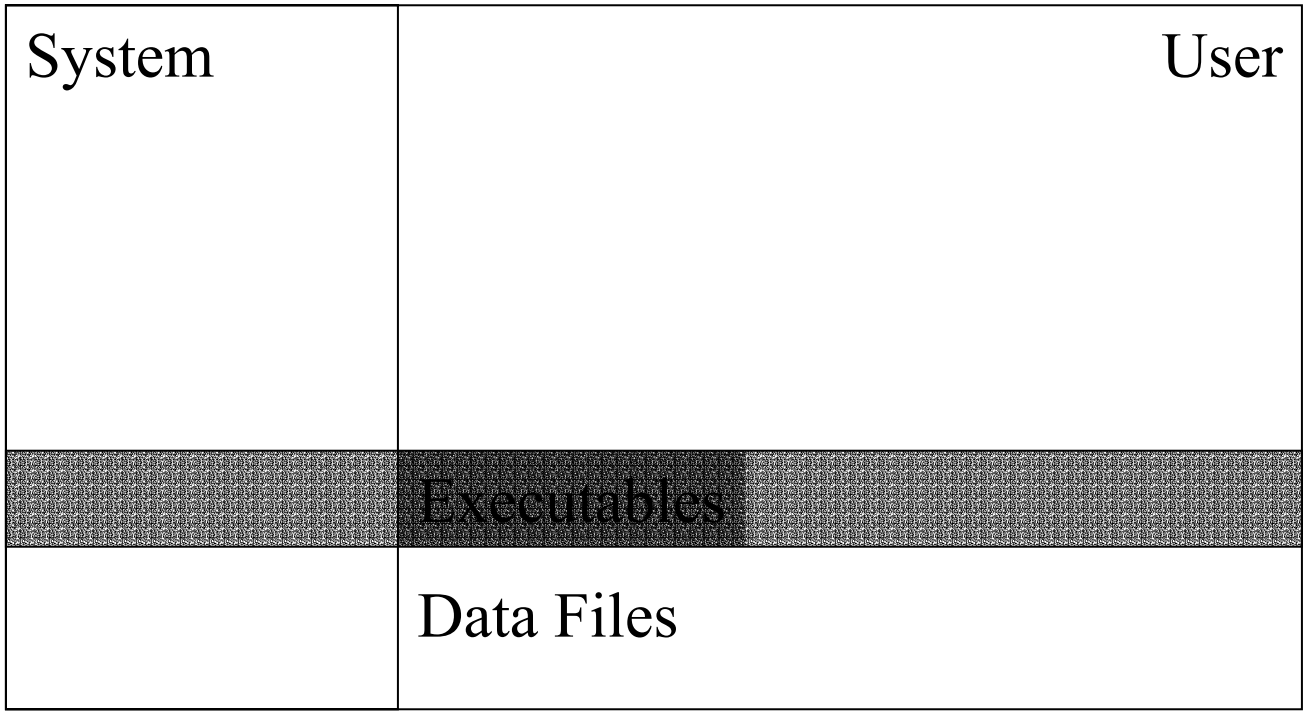
real_msgsnd Exposure



verify_area Exposure



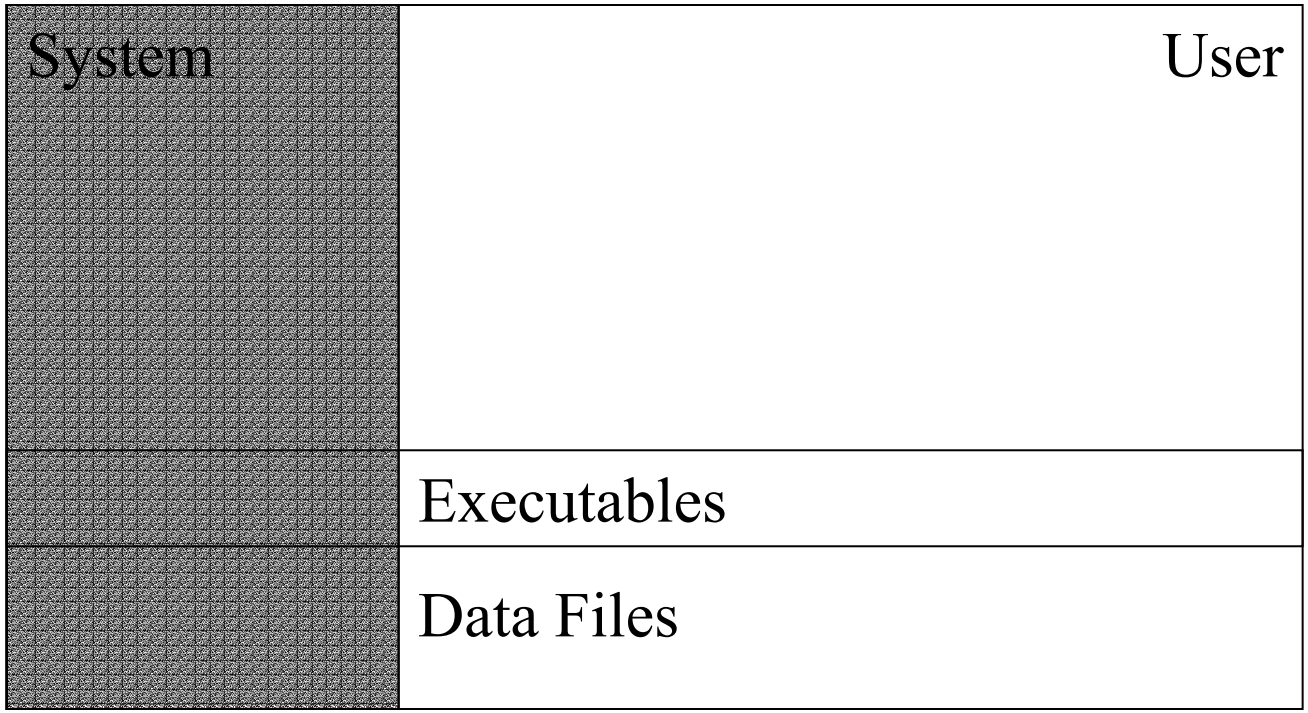
Non-Readable file *ptrace* Vulnerability



Total Control of System

System	User
	Executables
	Data Files

Impossible in Traditional System



inode.i_count Overflow

System	User
	Executables
	Data Files

Conclusion

- A security vulnerability is not an “all or nothing” proposition.
- There are various levels of security degradation that fall between an adversary gaining total control of the system and him having no effect at all.
- We have presented quantitative ways to measure flaw severity and these levels of degradation.
- These are the first such metrics which fulfill the need to measure, quantify, and compare various flaws.

Department of
Computer Science

Predicting the Severity of Intrusion Series

- Motivation for the work
- Analysis
- Conclusions and Future Work

"A single intrusion is a tragedy. A million intrusions is a statistic."

Motivation

- Are over 90% of the security incidents due to known problems?
- Anecdotally true, but how do we provide stronger evidence?
  - Perform an analysis of past intrusions using the CERT/CC™ historical database.

Data Collection Procedure

- Search CERT summary records for key words and vulnerability number (automated).
- Review summary record and electronic mail to ensure valid (manual).
  - If evidence didn't support the fact that an intrusion took place, then the record was not counted (results in an under count).

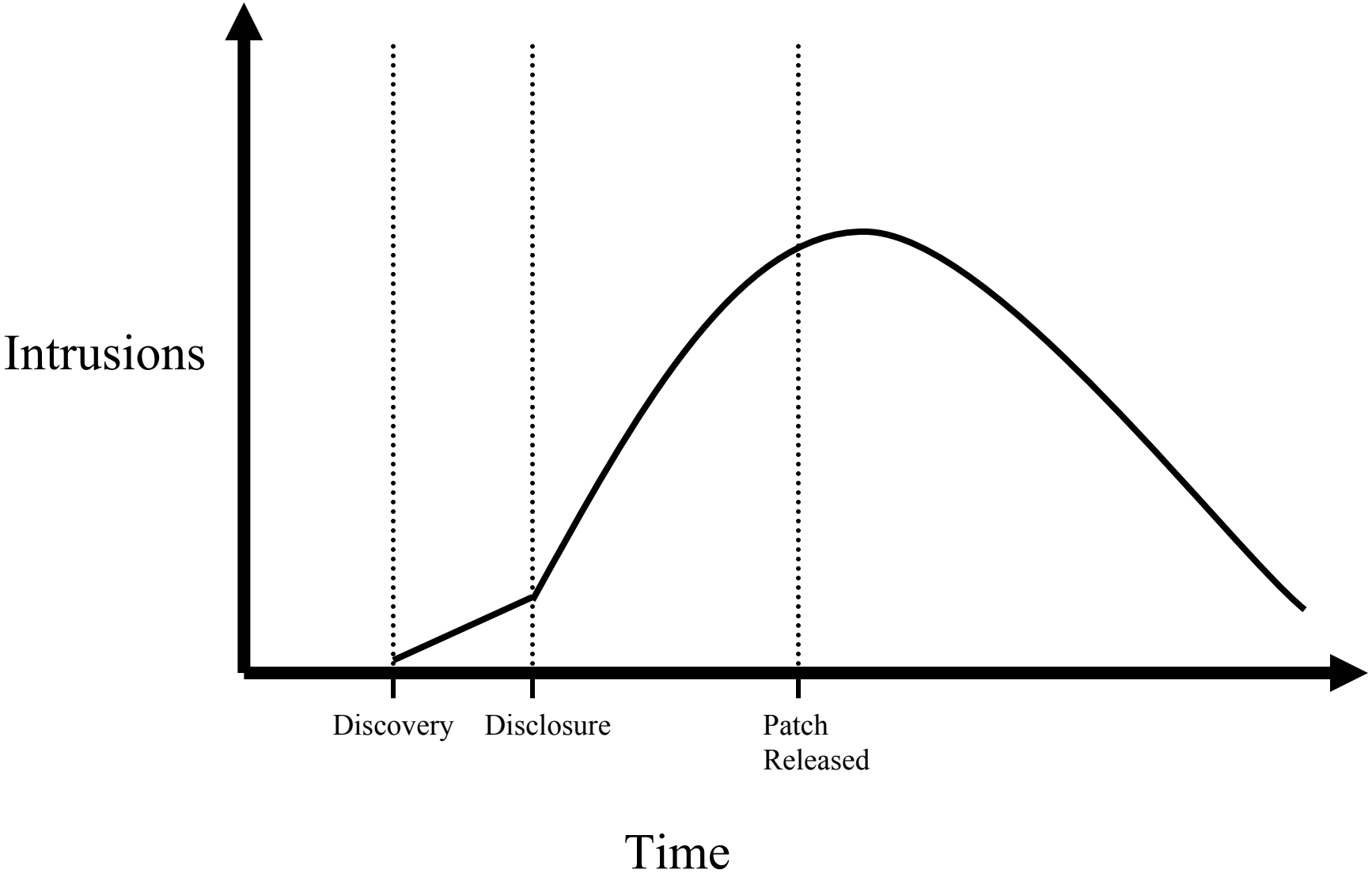
CERT Data Issues

- Intrusion reports are self-selecting.
- People can't report what they don't know or understand.
- Human element
  - Errors
  - Boredom
- Until recently records were not conducive to analysis.

What We Expected to find

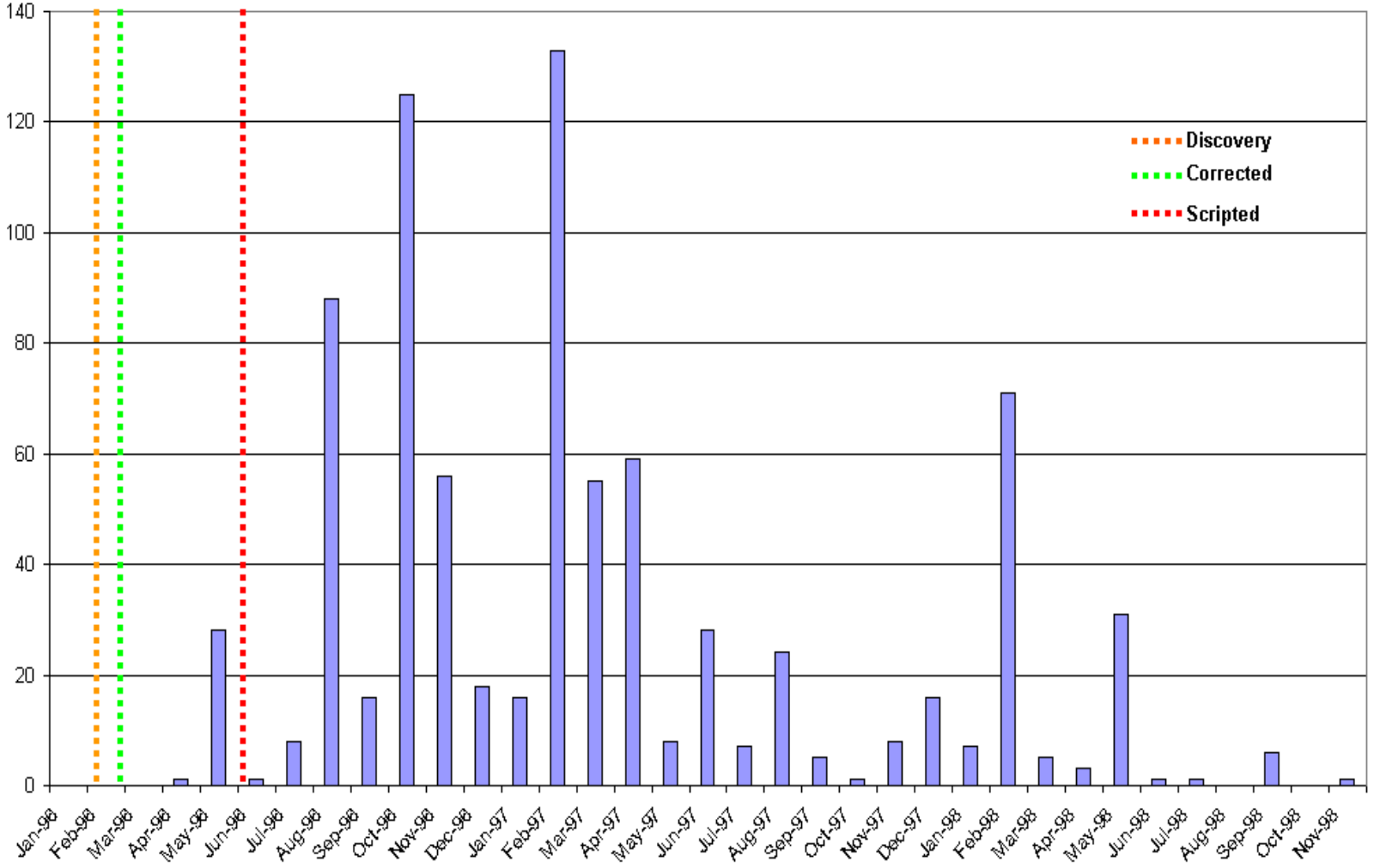
Wasn’t there

Intuitively



Department of
Computer Science

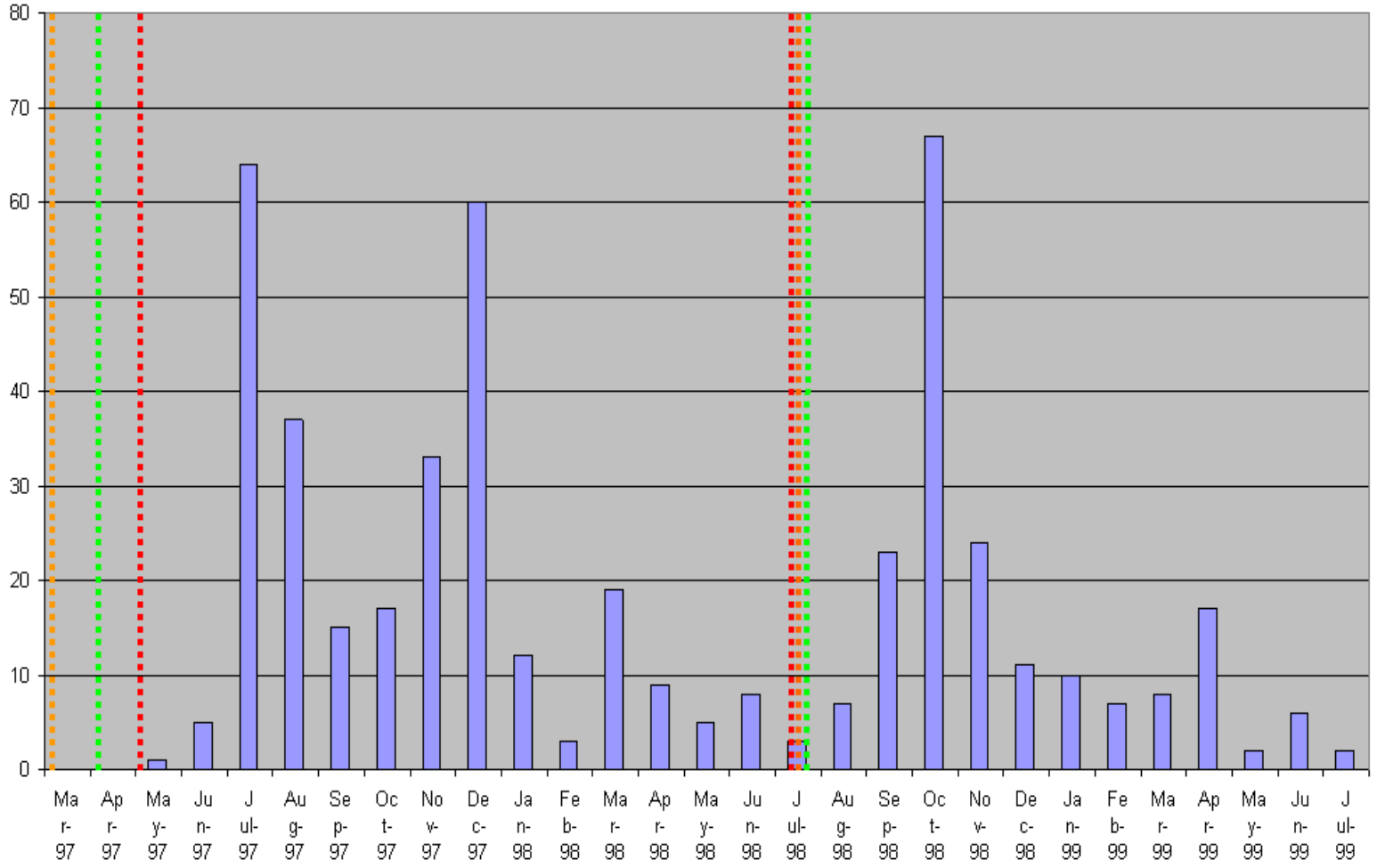
Intrusions due to phf exploit[1]



[1]IEEE Computer Magazine, December 2000, Vol. 33, No. 12, pp. 52 –59.

Department of
Computer Science

Intrusions due to IMAP exploits[1]

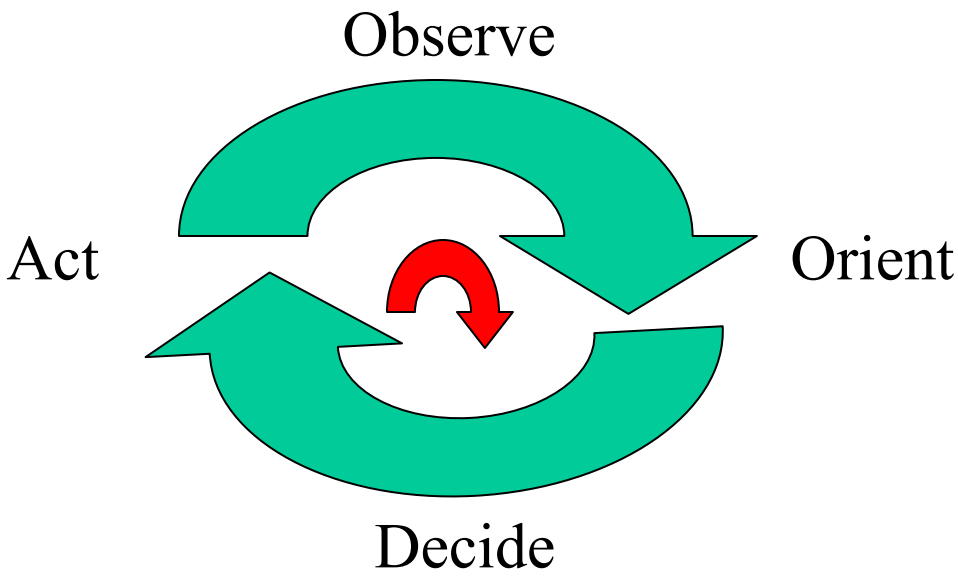


[1]IEEE Computer Magazine, December 2000, Vol. 33, No. 12, pp. 52 –59.

CERT data supports the hypothesis

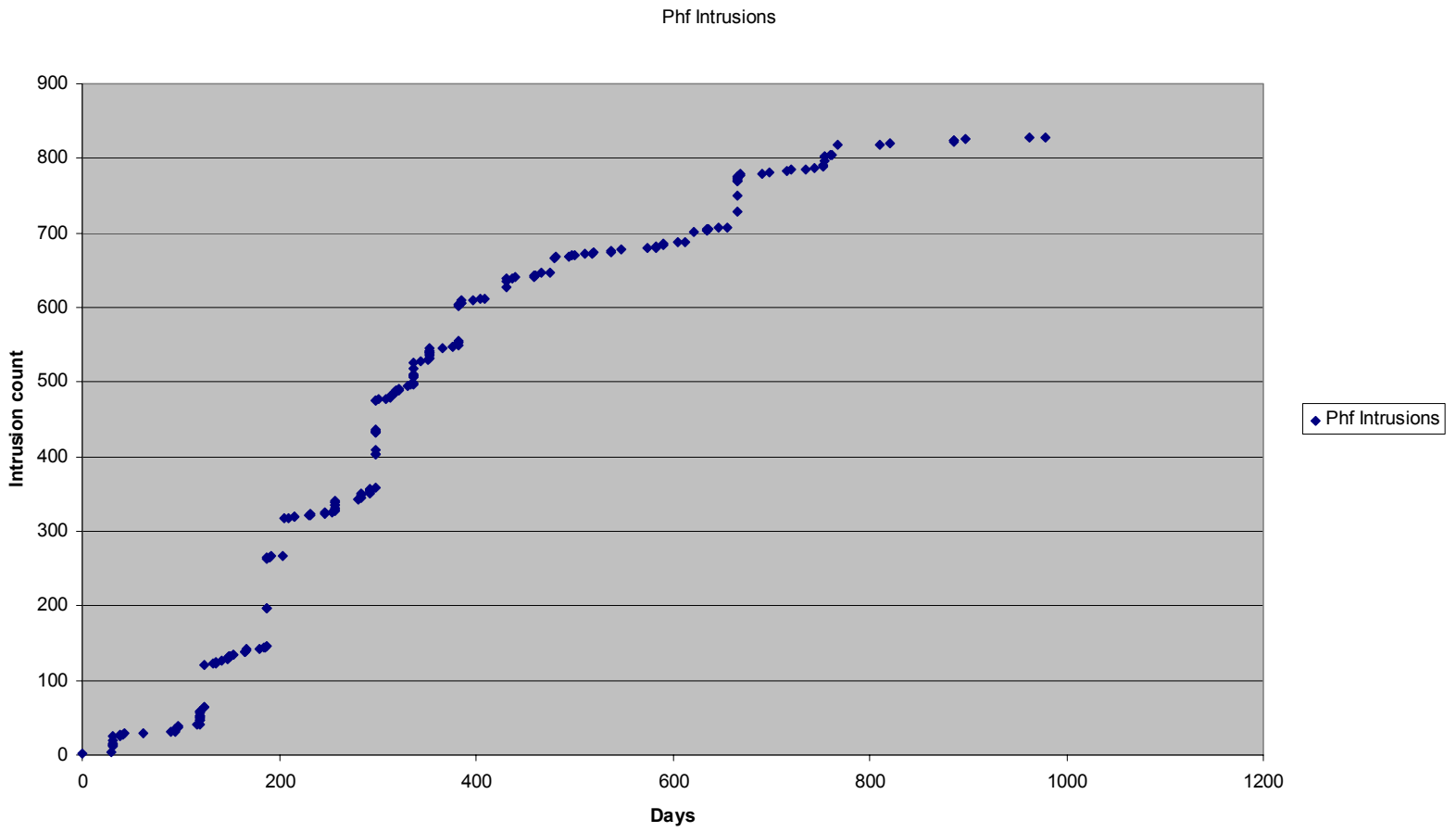
- Well over 90% of the security incidents reported to CERT could be prevented!
- Attackers have automated (scripting) and as a result react faster than the defenders!

Attackers are within the defenders decision loop.



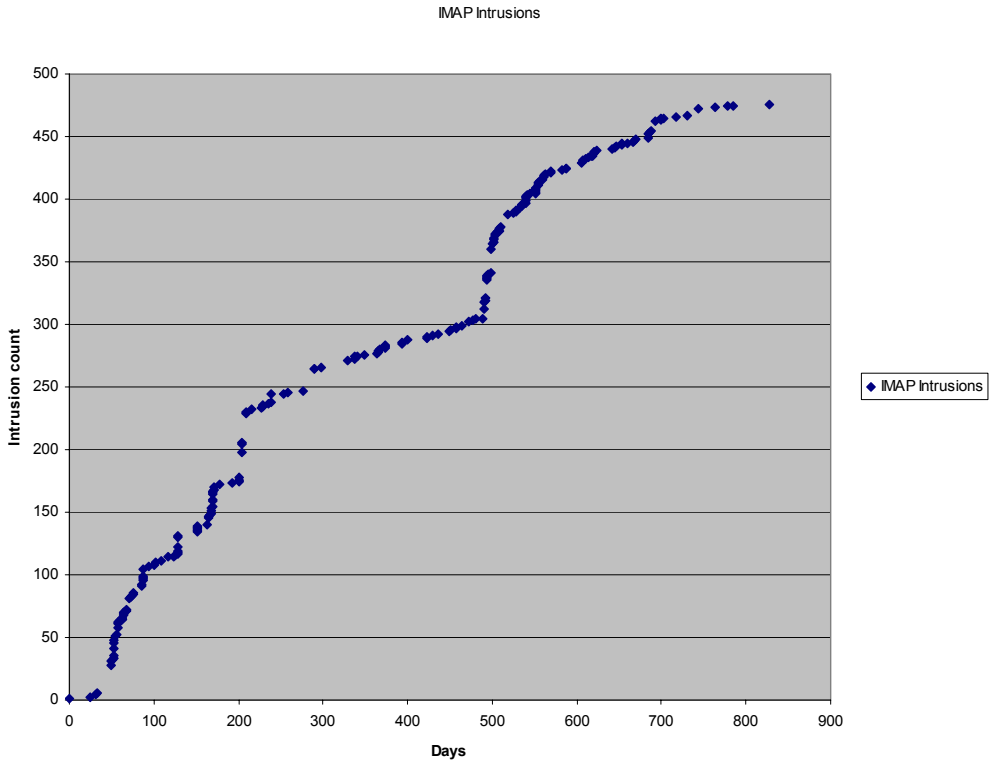
Something Entirely Different

- Analysis of several incident histograms indicated that the intrusions accumulated with a similar shape.



Was this just a fluke?

- Perform a linear regression analysis and collect more data to see.

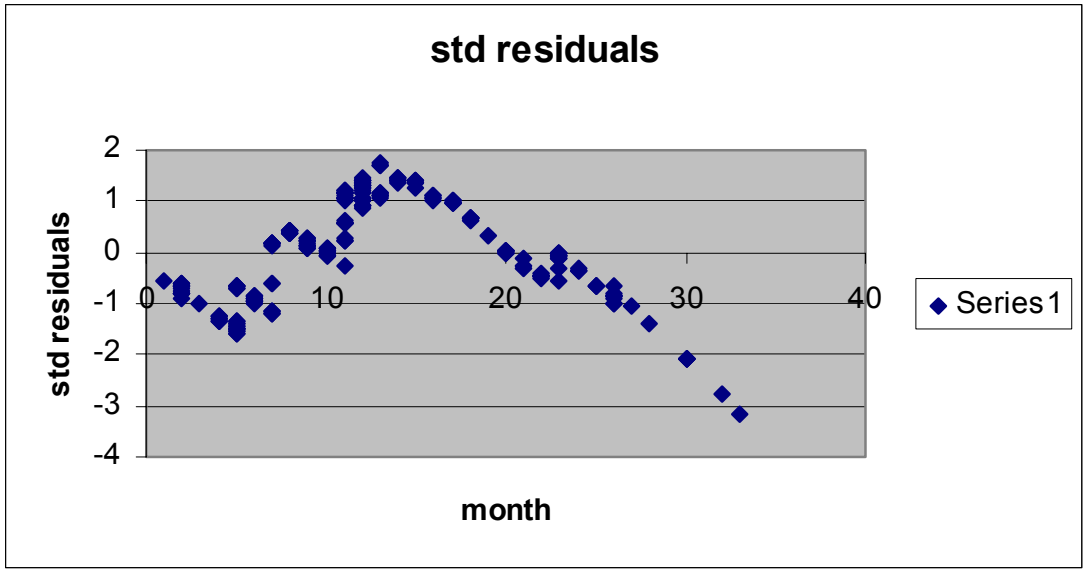


Can We Predict the Severity?

- If we can find a model that fits, then we may be able to predict the severity of incidents.
  - NOTE: We are ONLY curve fitting. We are not making statements about any potential relationship between the independent and the dependent variables.
  - We focus only on the slope found from the regression analysis.

Why only a curve fit?

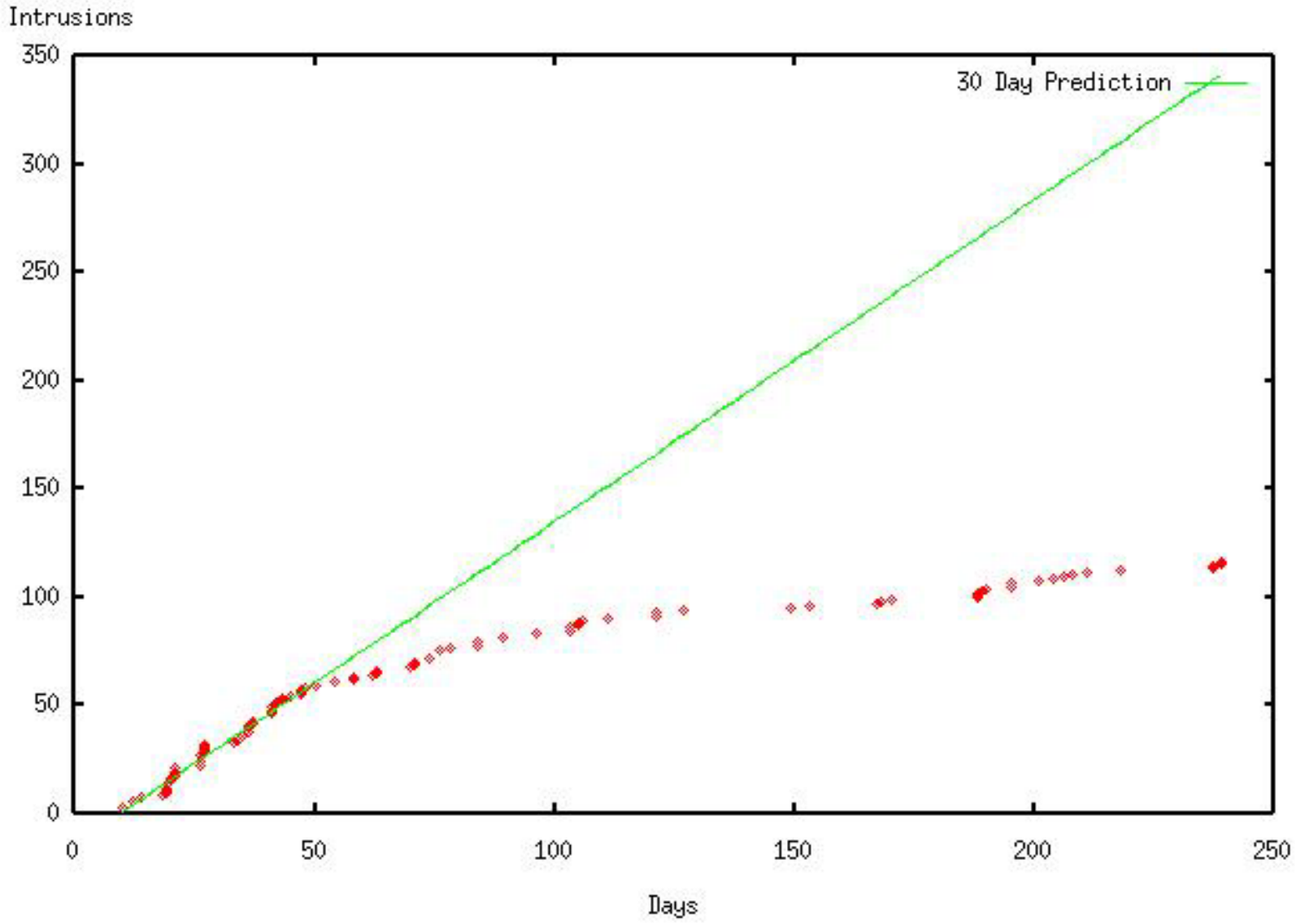
- Biases in data
- Accumulation function is linear in nature
- Residual plots (phf shown)



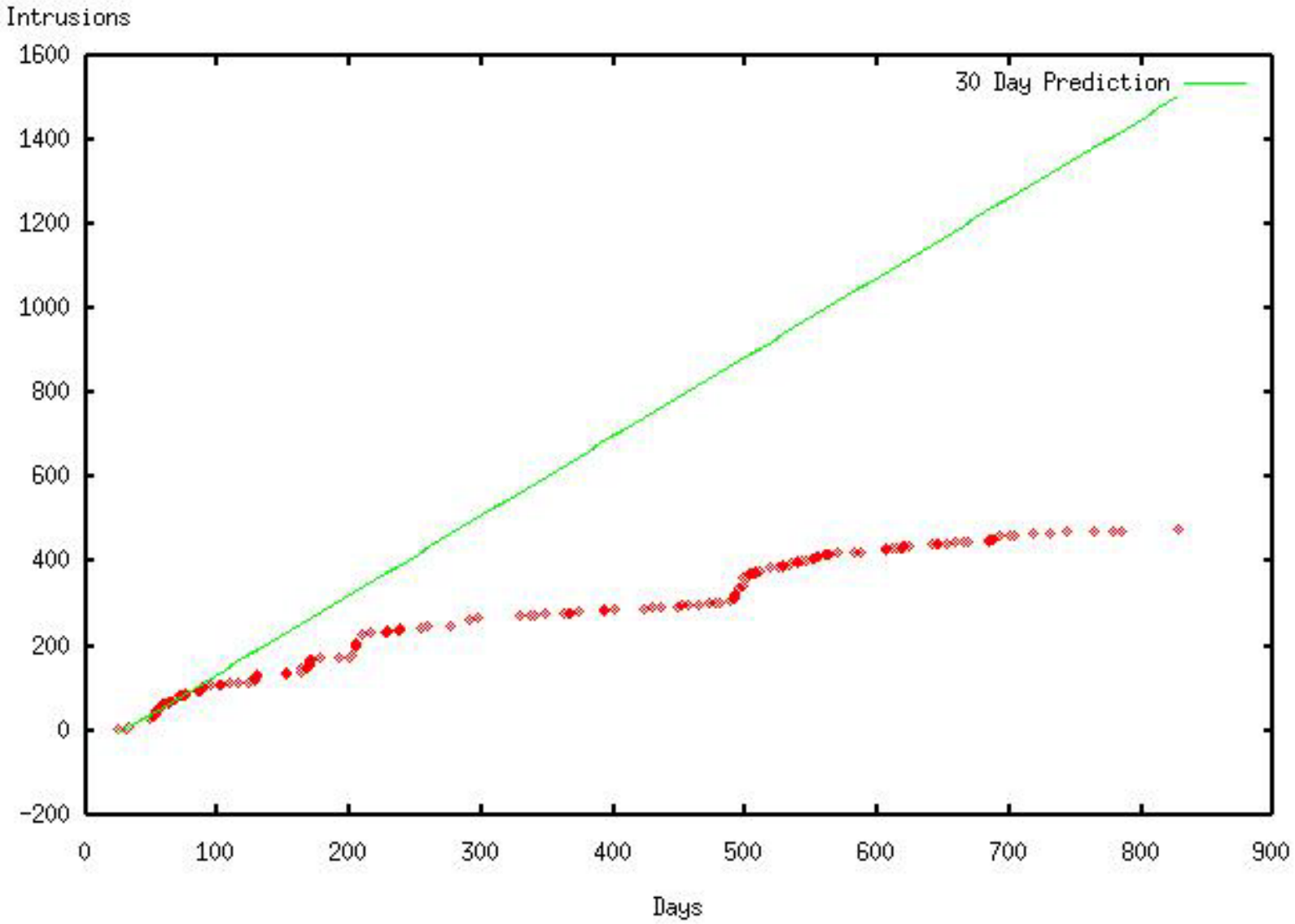
Promising Approaches

- Initial analysis focused on examining the data on a monthly basis. Demonstrated useful results but …
  - Introduced a basis (not all months are of equal length)
  - Prediction not useful after three months
- Looking at a daily analysis now
  - Regression done after 30 days of activity

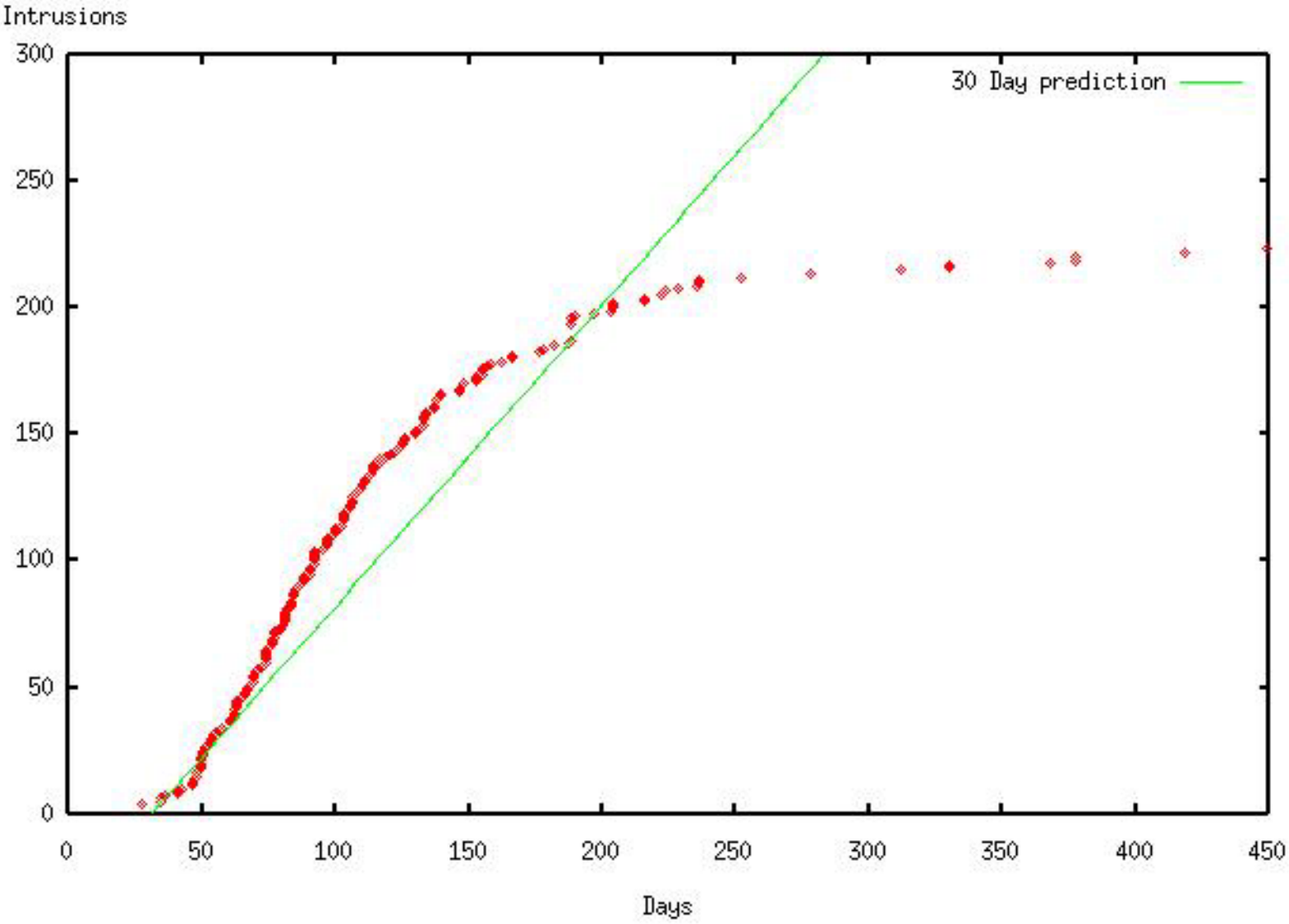
statd format



IMAP


Intrusions
1600
1400
1200
1000
800
600
400
200
0
-200
0
100
200
300
400
500
600
700
800
900
Days
30 Day Prediction

wu-ftpd


Intrusions
300
250
200
150
100
50
0
0
50
100
150
200
250
300
350
400
450
Days
30 Day prediction

Over twenty years of Security Research?

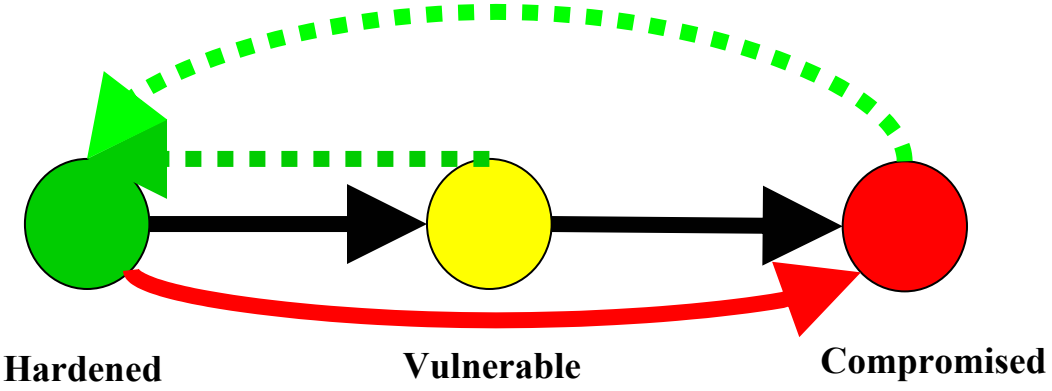
- Yet- wide-spread intrusions occur daily in all types of organizations!
- Perhaps rather than focusing on the technology for secure systems- we should focus on the technology for the management of systems securely?
  - Strong Configuration Management
  - Automatic Patch Installation
  - Exploitation Detection
  - Recovery and Reconstitution

Our Approach

- Understand and Formalize the Problem
- Develop a “ground” for Trust
- Automate

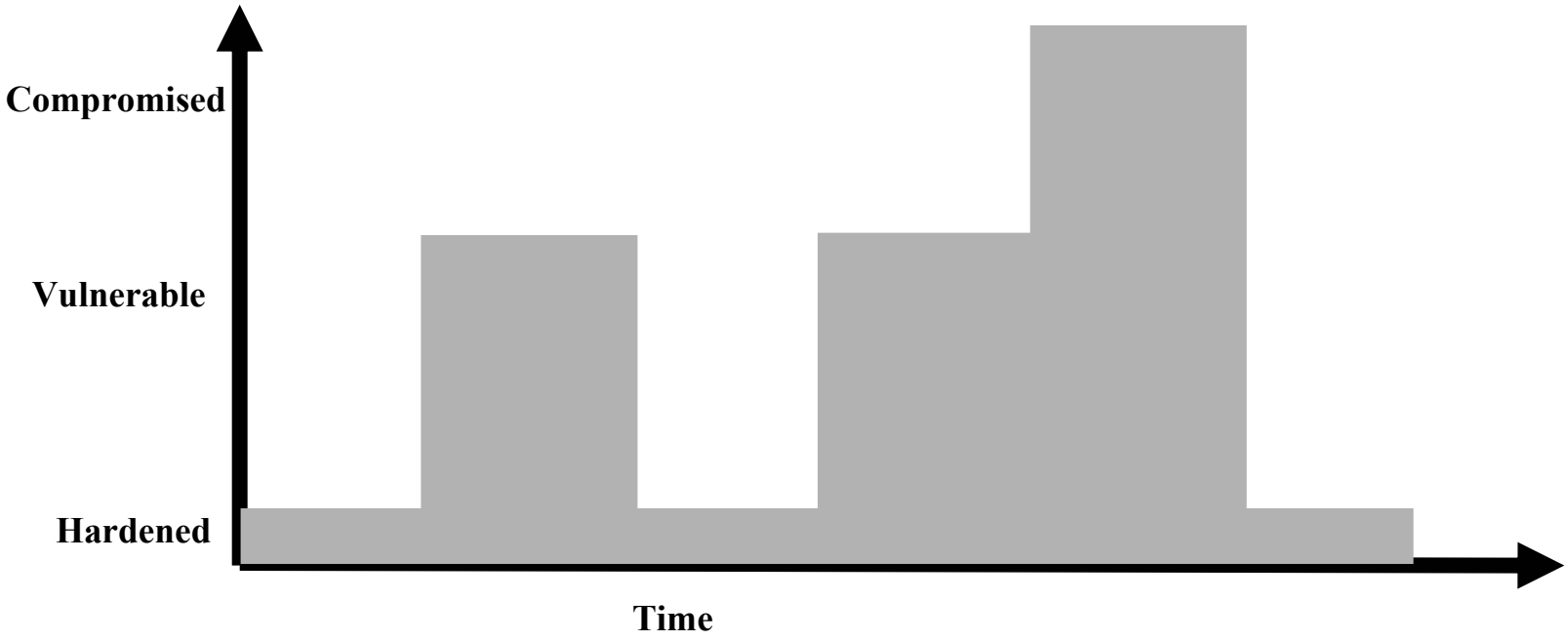
Understanding the Problem

•Model the life-cycle of an information system based on a state system.



Defining the Window of Vulnerability

•The Window of Vulnerability is the sum of the total time that a system is vulnerable to a known exploitation, and the total time that a system is compromised.



Active Systems Management

- Goal is to shrink the Window of Vulnerability to as small as possible.
- The attackers have automated- the defenders must as well!
  - Komoku
  - wBox

Department of
Computer Science

Komoku – An embedded Trust "ground"

- Security and Management applications are inherently un-trusted?
  - Why? Because they rely on the validity of the operating system?
  - What if the operating system is compromised?

- Komoku is an embedded co-processor (possibly tamper protected) which can:
  - Perform real-time integrity checks through active IO requests or passively by "snooping the IO bus"
  - Perform secure configuration systems management
  - Perform incident post-mortem analysis and recovery

wBox

- Wireless networks are quickly becoming ubiquitous much like Internet connections many years ago
- Much like Internet connections before firewalls- wireless access points (AP) may provide an attacker access to your internal network:
  - Access control for wireless networks is non-existent
  - WEP v1.0 has serious weaknesses
- wBox acts as an access and security manager for wireless networks (joint work with Narendar Shankar and Justin Wan):
  - Dynamic WEP key management via DHCP interface
  - IPSec, packet filtering, and intrusion detection capable

Conclusions

- The security problem is worse than most suspect.
- The attackers have automated, but the defenders have not!
- Improving security and systems management appears as the area with the greatest potential impact.
- Automation with a trust “ground” is the key.

Future Work?

- Working with statistician to gain greater insight:
  - Grouping data better
  - Multivariate regression
- Start analysis from scripting date
- Continuing to collect more data
- Focusing on methods to tighten the defenders decision loop