A Case Study on Constructing a Security Event Management (SEM) System

Alcatel·Lucent

Vijay K. Gurbani
Joint work with D.L. Cook, L.E. Menten, and T.B. Reddington
Security Technology Research
Bell Laboratories, Alcatel-Lucent
vkg@{alcatel-lucent.com,bell-labs.com}

3rd International Conference on IT-Incident Management and IT-Forensics (IMF 2007), Universität Stuttgart
Stuttgart, Germany
September 13th, 2007

Agenda

- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

Cacker vs. defender

Only needs to find **one** weak element

Relies on fact that protection is not perfect

May be as knowledgeable (and more so) as the defender

Needs to protect **all** elements

Needs to be perfect in

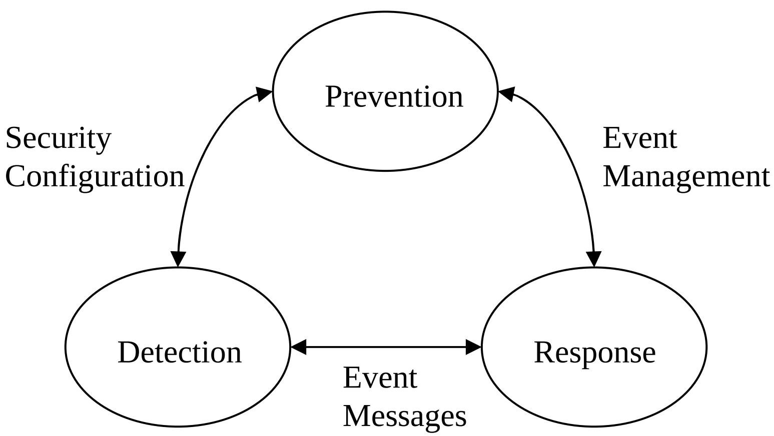
- Design
- Implementation
- Configuration
- Operations

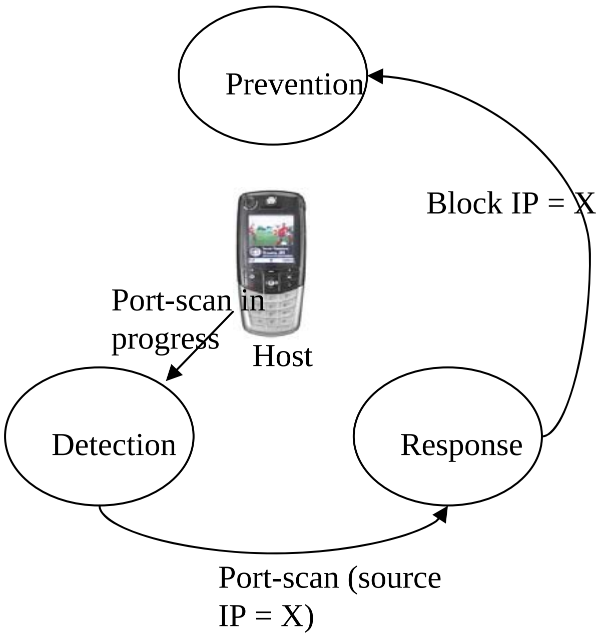
It will be a long time before designs, implementations, configurations, and operations become perfect.

It is far easier to be a cacker than a defender!

Introduction

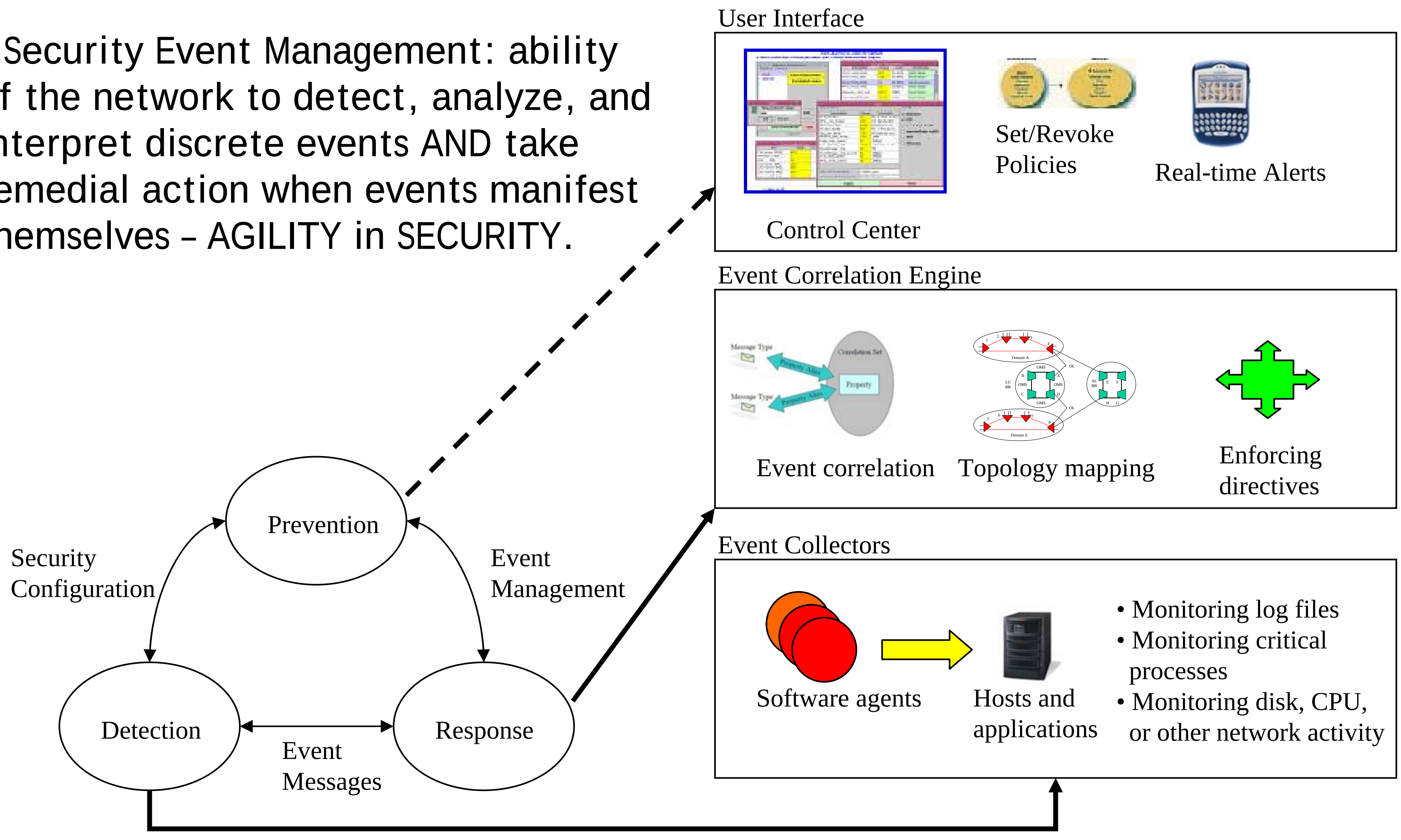
- Why is network security difficult?
  - Delegated to individual hosts and applications;
  - Limited communications with others in the ecosystem;
  - Applied in *reactive* mode, not *pro-active* mode;
  - Network security tools do not provide an integrated network view regarding the state of the network: minimal, if any, situational awareness.
- What should the model be?





Introduction

- Security Event Management: ability of the network to detect, analyze, and interpret discrete events AND take remedial action when events manifest themselves – AGILITY in SECURITY.



- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

Alcatel·Lucent

Related work

- SEM frameworks
  - Commercial
  - Academic: SEM using data-mining techniques:
    - Liu et al. – SEM system constructed using CASE-based reasoning.
    - Ertoz et al. – MINDS – Minnesota Intrusion Detection System.
- Determining root-cause analysis
  - Duan et al., Sekar et al.: enhance IDS to minimize false alarm rate.
  - Julisch: few dozen root causes account for >= 90% of alarms an IDS generates.
  - Devit et al.: topological proximity approach to wean out implausible alarms.
- Reporting
  - Debar et al. [RFC4765]: IDMEF - describes a data model to represent information exported by a IDS for consummation by a response systems and management systems.
  - Feinstein et al. [RFC4767]: IDXP – an application level protocol for exchanging data between intrusion detection entities.
  - Mitre's Common Event Expression (CEE): establishes consistent log formats and terminology.

- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

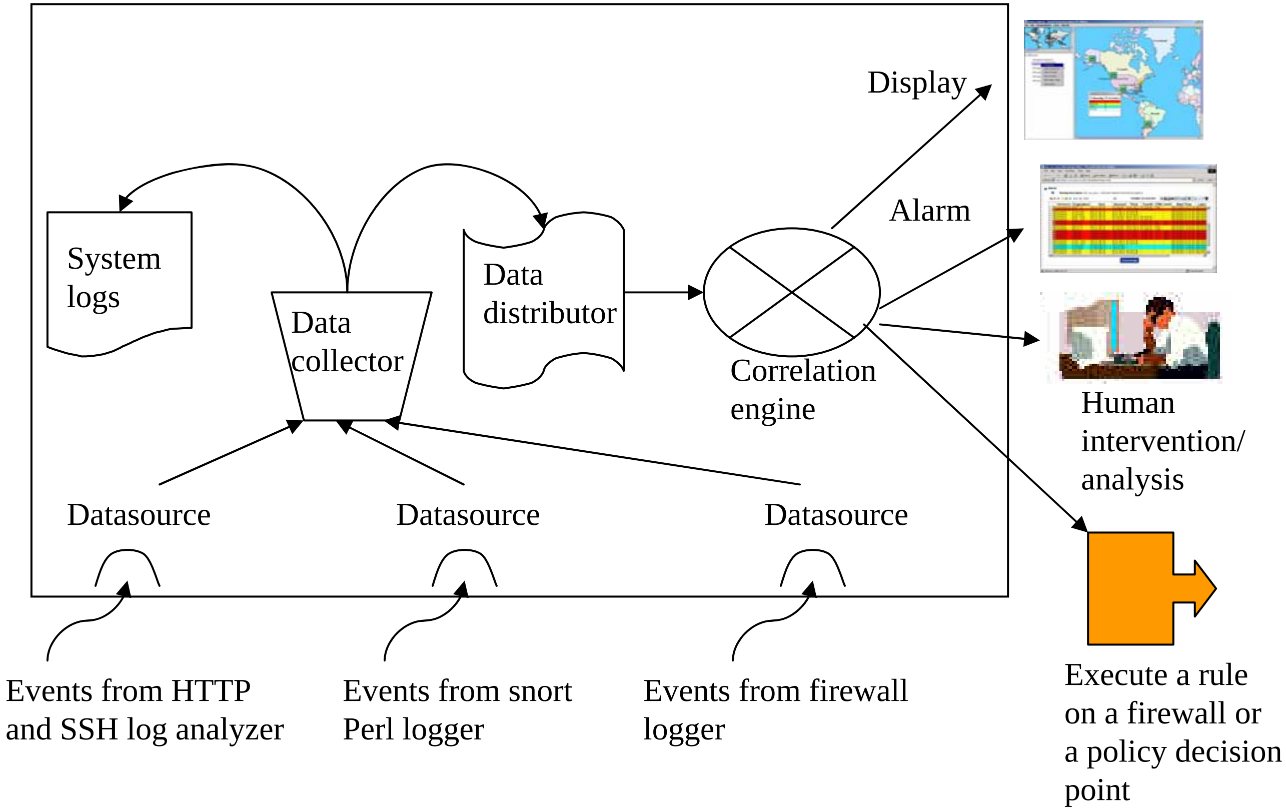
Alcatel·Lucent

System design and architecture

Tools and technologies used:

- Event correlation engine: Bell Labs/Alcatel-Lucent correlation engine used in fault management systems.
- Intrusion detecting systems used:
  - Open source: snort
  - Bell Labs: HTTP CLF and SSH LF analyzer
    - HTTP: >= 2000 attacks from CVE dictionary;
    - Statistical inference module: triggered on inter-arrival time, errors generated, or links accessed;
    - Exponential weighting module: if (flow utilizing >= 75% link bandwidth) drop, diffserv, …
    - SSH login failed attempts
- Bell Labs: Filesystem integrity checker
- Load generators:
  - Open source: Nikto – web load generator, nmap, and snort
- Firewall: Bell Labs/Alcatel-Lucent firewall providing session establishment rate limiting, traffic rate limiting, IP address/header inspection, etc.

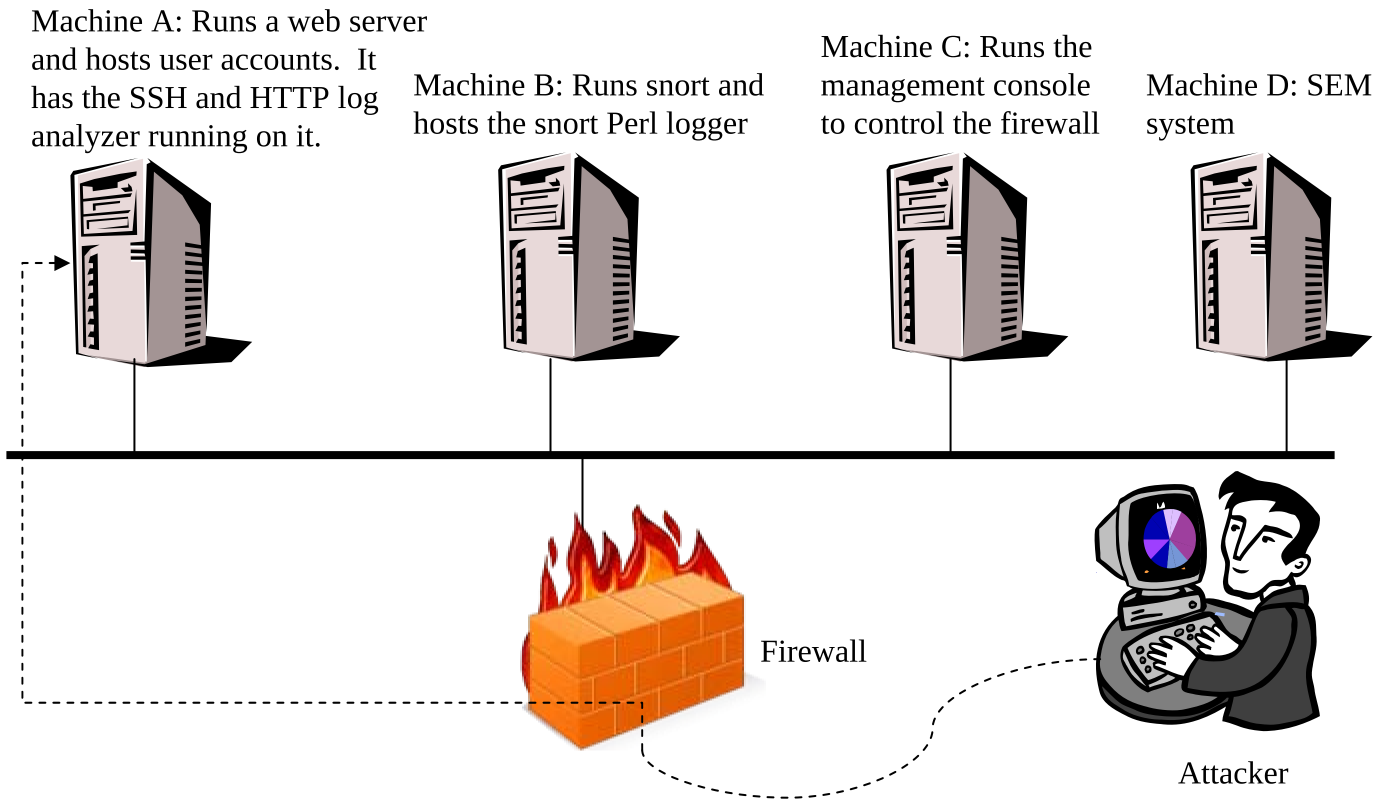
System design and architecture



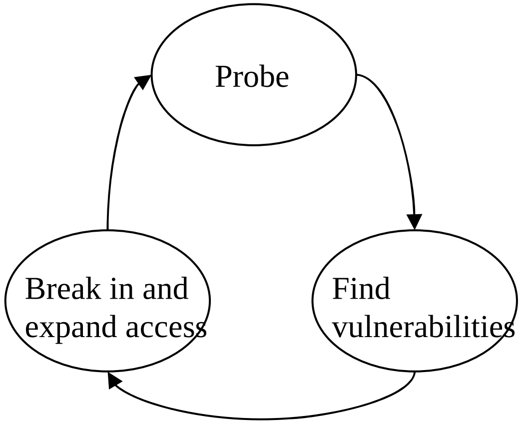
- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

Alcatel·Lucent

Using the SEM framework

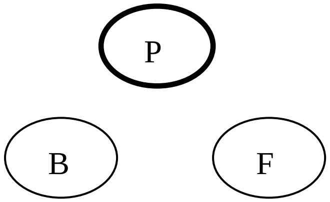


Using the SEM framework



Typical attack cycle

Stage 1: Network reconnaissance



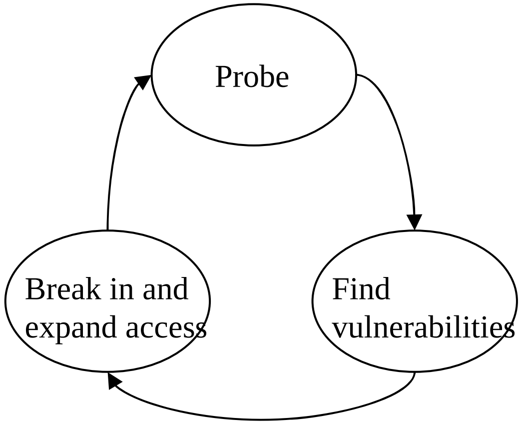
Network scan by nmap.

Events generated: snort, firewall (incoming packet rate-limit violation)

Correlation done to associate these events to be part of the same attack.

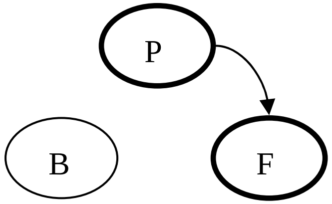
Remedial action: Issue denunciation to block offending IP.

Using the SEM framework



Typical attack cycle

Stage 2: Find and exploit vulnerabilities



1. Machine A targeted.

   HTTP exploitation using nikto.

   Events generated: Bell Labs HTTP CLF generator.

   Remedial action: Issue denunciation to block offending IP.

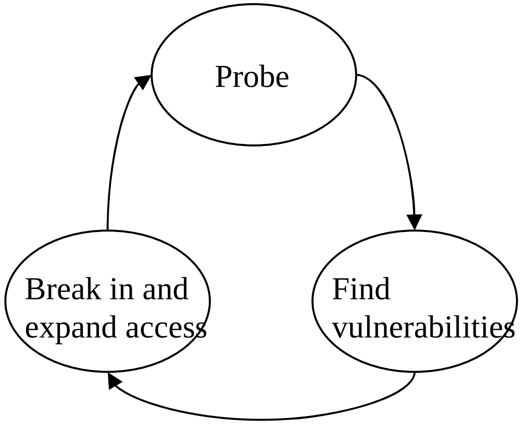
2. Continue port scans on A.

   Discover a server on a high port.

   Attack the server (buffer overflow, deus ex machina)

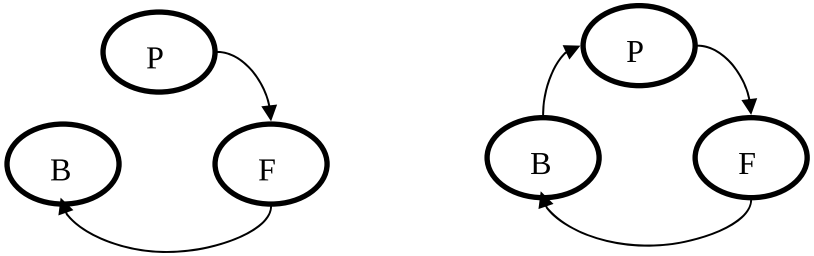
   Get password file; run password cracking program.

Using the SEM framework



Typical attack cycle

Stage 3: Break-in and expand access



1. Machine A compromised.

   Attacker logs in using ssh; no event generated.

   Attacker uses *wget* to fetch a trojan program; no event generated.

   Executes trojan; makes outbound connection.

   Events generated: snort outbound connection.

   Rule fired: No non-HTTP outbound connections allowed.

   Mitigation action: File integrity check on user.

AT THIS POINT, HUMAN INTERVENTION REQUIRED. ATTACK SUCCEEDED!

- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

Lessons learned

1. Network fault management complements, but is different than SEM.

   Can a NFM be transformed to a SEM? No, not quite.

   Root-cause analysis is different.

   NFM:

   - Appropriate solution to address problem is valid once root cause is found.
   - Root cause is normally a single component in a fixed geographical location.
   - Damage caused typically does not escalate with time.

   SEM:

   - Attack proceeds in phases; characterized by [1…N] sources accessing [1…M] destinations, cardinality of N,M is disjoint.
   - Root cause proceeds from P->F->B and the cycle repeats with another element in [1…M].
   - Appropriate solution is dependent on current root cause.

Lessons learned

Can a NFM be transformed to a SEM? No, not quite.

Network attacks are dynamic.

NFM:

- Many categories of root causes, but small and predictable set of events for identifying root causes.
- Threshold settings relatively stable, and correlation rules change infrequently and are a function of network topology, which is generally static.

SEM:

- Few categories of root causes, but a great many and unpredictable set of events.
- Threshold settings will vary, making it hard to derive static correlation rules.

Lessons learned

2. Event thresholding, correlation and mitigation must be pushed to the edges.
   - Event flooding is a problem for central correlation.
   - Use P2P techniques, not to search, but to efficiently divide the space.
3. Security event records must be designed for SEM consumption.
   - Firewalls produce many records for a session, but these records did not have a flow label.
   - Establish CLF for protocols – HTTP has one, why doesn't SSH? SIP? SMTP?
4. Adaptive remediation strategies.
   - Network operators are reluctant of automatic policy control.
   - Example: instead of dropping HTTP traffic, redirect to a honeypot.

- Introduction
- Related work
- System design and architecture
- Using the SEM framework
- Lessons learned
- Research agenda in SEM
- Open floor

Research agenda in SEM

Today: Building a SEM is a task in integration and "glue programming."

- No formal language from SEM to control or query edge devices.
- No formal language from edge devices to SEM for reporting.

Research plan:

- Better network reconnaissance techniques.
  - Today's focus is on DDoS attacks => lot's of events generated.
  - Can we detect a cracker that has created zombies on your network and logs into the master zombie server to issue a *1-character command*?
- Develop resilient protocols.
  - Ironically, it is precisely when a network is under attack that it is least able to devote bandwidth resource for informing a SEM system.
  - "Parsimonious Protocols": idempotent, self-contained, minimal retransmissions and ACKs – 20% packet loss, 5 copies of a message sent ➔ 99.6% probability that at least one copy will get through.
  - From SEM to edge devices, the protocol must be more than a "TCP connection".
  - Standardization? Perhaps.

Research agenda in SEM

- Policy language and rule-based systems.
  - What information should be collected by edge devices? How?
  - Can anomaly detection be better done through rule-based systems (AI)?
- Device modeling.
  - How to provide SEM with characteristics of each controlled device? Location of each controlled device? Can a device "learn" from the events so it only reports events of interest to the SEM?
- The effect of network topology on correlation rules.
  - Specifics about network topology is embedded in rules and actions encoded in a SEM system. Will changing the network topology break these rules? Can the ruleset be automatically changed to allow for a topology reconfigaration?
- Integration with OAM&P.
  - Many SEM rules end up modifying an ACL at a traffic control point because *many* suspect events occur in a short timeframe.
  - What if there was *one* event that crippled your network service?
- Developing HCI for security (HCISec).
  - Multidisciplinary approach for presenting and soliciting information to users.

Open floor

Thank you!

Vijay K. Gurbani
Bell Laboratories, Alcatel-Lucent
tel: +1 630 224-0216
mailto: vkg@bell-labs.com
vkg@alcatel-lucent.com

Alcatel·Lucent