An Introduction to Event Modeling and Correlation

Stephen Rondeau

Institute of Technology

Agenda

- Background
- Recording Events
- Event Operations
- Modeling Events
- Correlating Events
- Commercial Approaches
- Rule Based Correlation: SEC
- Conclusion

Background

- Expect computers and network devices to:
  - Do the functions we desire
  - Have good performance and adequate capacity
  - These criteria constitute the initial baseline
- Things are happening constantly
  - Services running (e.g., firewall, virus scanning, login)
  - User input processing (e.g., keyboard, mouse)
  - User output processing (e.g., screen updates)
  - Network handling (e.g. packet inspection and storage)
  - OS operation (e.g., paging, file management)
- 1000 to 1,000,000+ things per day, depending on:
  - volume of processing/device
  - number of devices in managed network

Background (cont.)

- “Things that happen” are events
  - Come from OS, IDS, services, applications, database, computer/network hardware monitors, user activity
  - Often indicate change of state
  - A message describing event may be recorded
  - Vary in importance from informational to critical
- Normal events are expected
- Abnormal events are unexpected
  - Includes missing events

Events Examples

- **Linux Syslog**

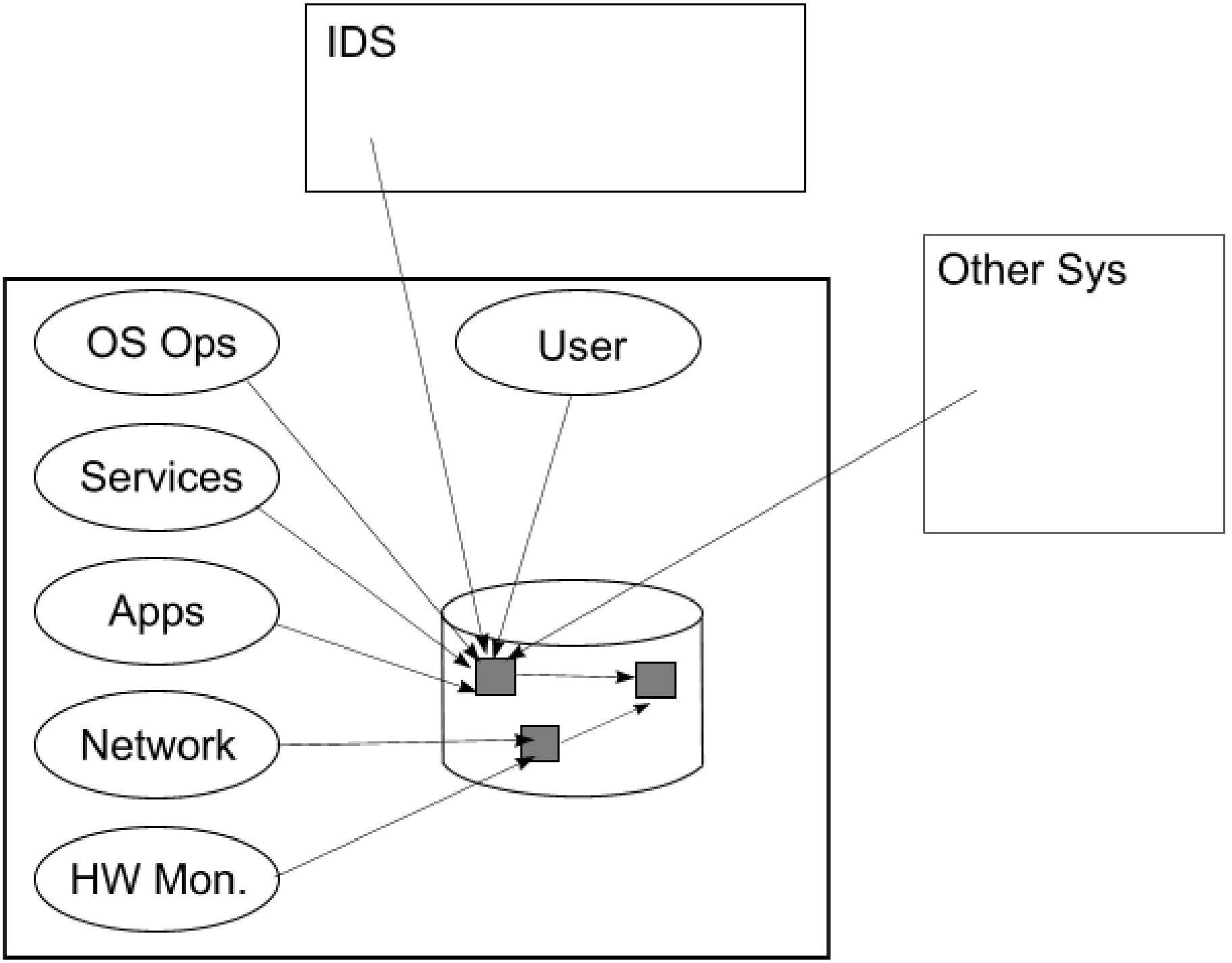
Feb 12 04:19:34 consensus ntpd[1921]: time reset +0.808076 s
Feb 12 04:26:01 consensus ntpd[1921]: synchronized to 140.142.1.8, stratum 2
Feb 12 13:12:09 consensus syslogd 1.4.1: restart.
Feb 12 13:12:09 consensus kernel: klogd 1.4.1, log source = /proc/kmsg started.
Feb 12 13:12:09 consensus kernel: Linux version 2.6.17-1.2187_FC5smp (brewbuilder@hs20-bc2-2.build.redhat.com) (gcc version 4.1.1 20060525 (Red Hat 4.1.1-1)) #1 SMP Mon Sep 11 01:32:34 EDT 2006

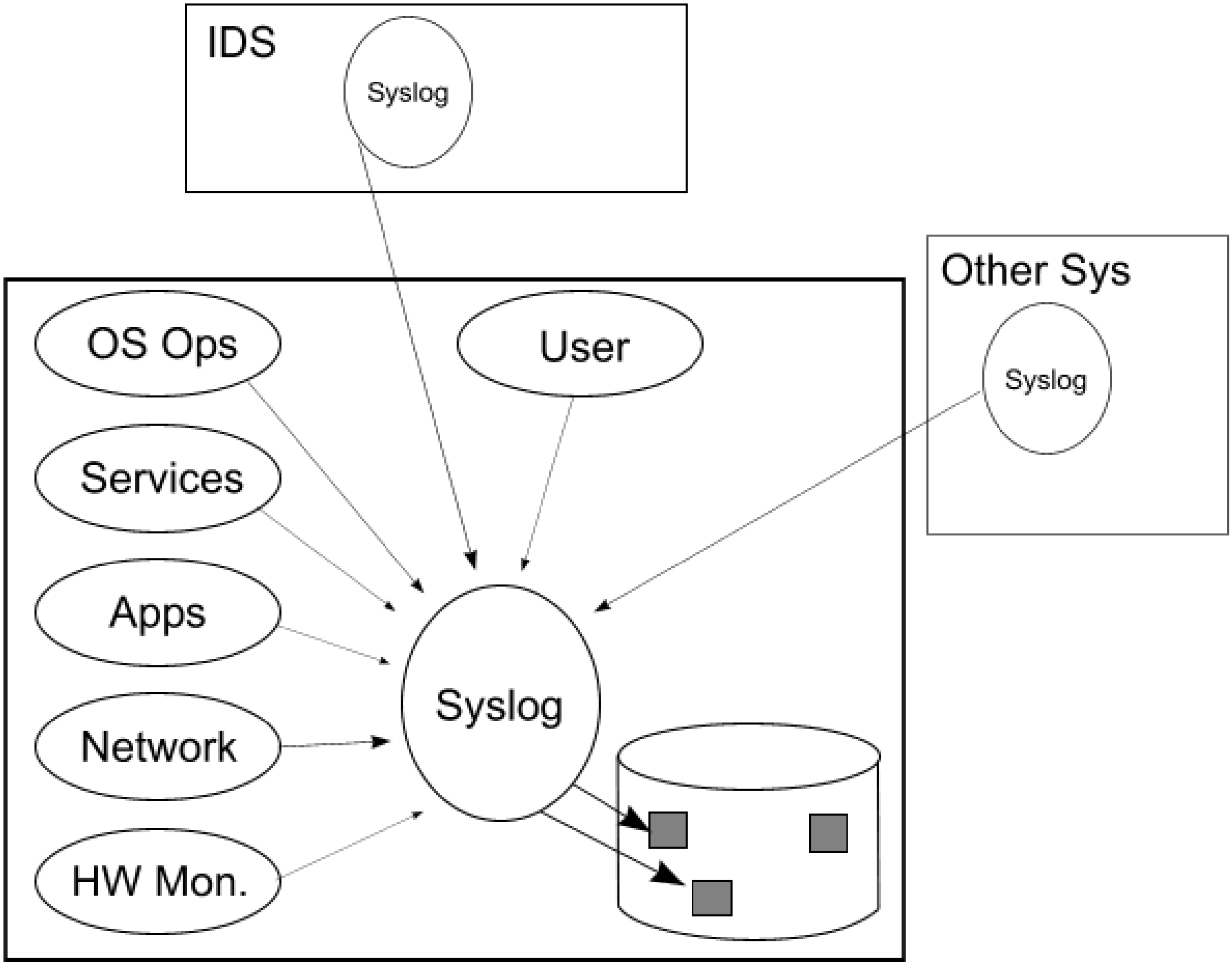
- **Windows EventLog**

Event Type: Failure Audit
Event Source: Security
Event Category: Account Logon
Event ID: 680
Date: 2/14/2007
Time: 4:26:32 PM
User: NT AUTHORITY\SYSTEM
Computer: AUTH1
Description:
Logon attempt by: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Logon account:joe
Source Workstation: \\WWW
Error Code: 0xC0000064

Recording Events

- Most events not recorded -- why?
  - Default: too many events
    - not enough time/space/people/expertise
  - No built-in mechanism to create event message
  - Mechanism exists, but not enabled
- Log files record event messages
  - Local or remote files
- Log files must be managed
  - May consume all storage
    - Could cause denial of service
  - Excessive information ignored; key events overlooked
- Log files can be processed online (real-time) or offline


IDS
Other Sys
OS Ops
User
Services
Apps
Network
HW Mon.


IDS
Syslog
Other Sys
Syslog
OS Ops
User
Services
Apps
Syslog
Network
HW Mon.

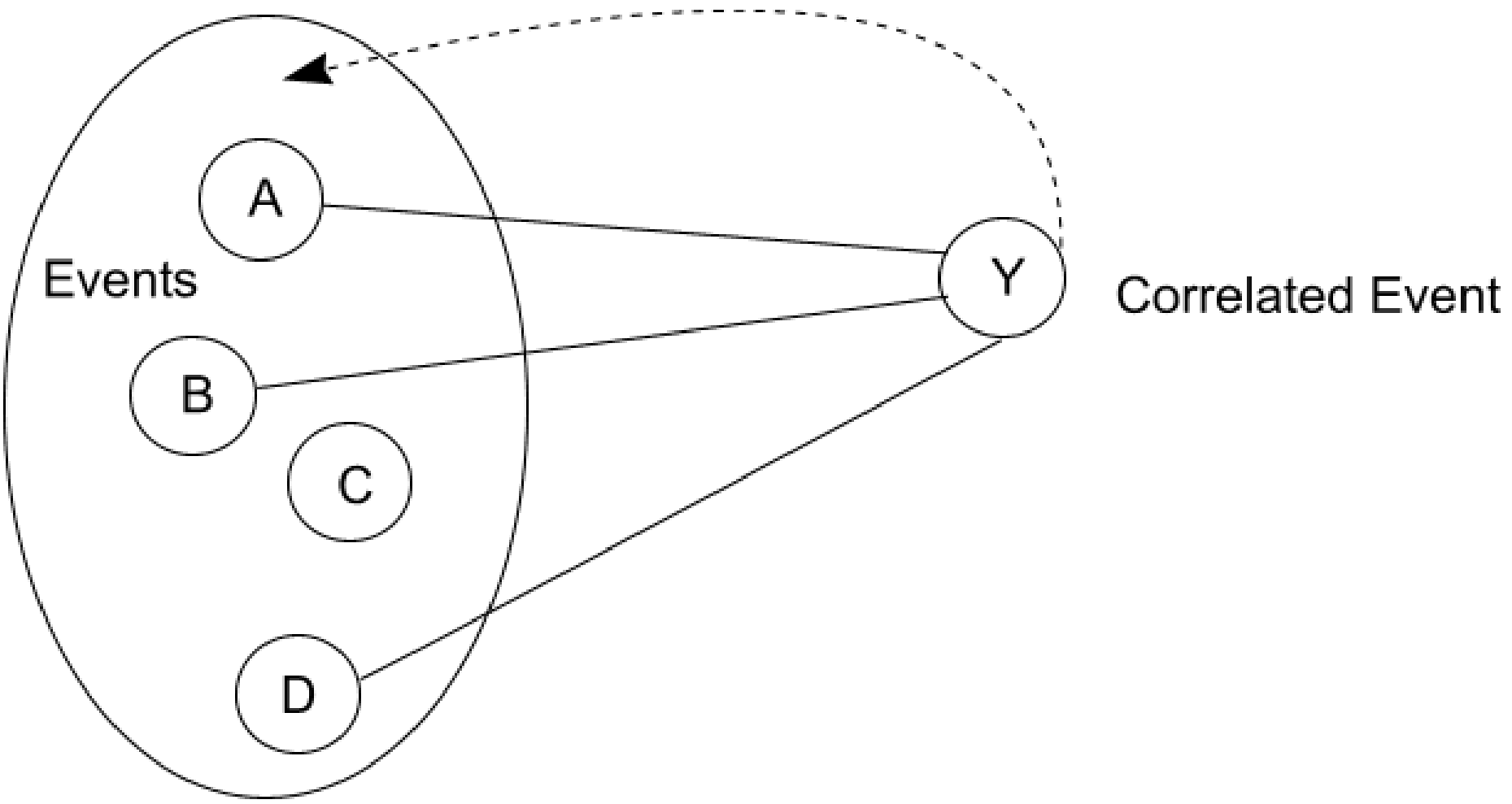
Recording Events (cont.)

- Not interested in all event messages
  - Only those that are the source or symptoms of problems
  - Only the first time a problem is reported, not every time
  - Maybe only those that occur a certain number of times, during a certain span of time, or both
  - Maybe only when an event is followed by a related event
  - Maybe only when a particular sequence of events occurs
- But how do you determine what is interesting? Later.
- Unix & Cisco syslogs; Windows EventLogs
- Rotate logs to reduce storage concerns
  - Overwrite oldest when size threshold reached
  - Keep n days, then overwrite oldest

Log File Monitoring vs. Correlation

- Many tools monitor logs for problems
  - LogWatch, LogSurfer, Swatch
  - rule: condition-> action: if event x occurs, then do y
    - x is interesting because it is in a rule
    - x must exist in the log files
  - Often analyzed well after the events have occurred
- Correlation: determine what happened; e.g,
    - Summarize sequence of events or record when number of events exceeds threshold by creating new event
    - Uninteresting events may be removed to reduce volume
    - Analyze logs: uncover patterns that will match events

Correlated Event



Event Operations

- **Filter**: select which events
- **Consolidate**: many events combined into one
- **Aggregate**: store events on some basis
- **Compress**: reduce number of similar events
- **Normalize**: convert to predefined form
- **Enrich**: add information to event
- **Generate**: tool creates new events
- **Correlate**: determine how to relate events

Examples of Detectable Incidents

- virus scanner turned off
- same alerts from Intrusion Detection System (IDS)
- login message with failed password message
- fast-growing disk consumption or network traffic
- many network ports being scanned from same IP
- many logins during off-hours
- multiple accounts failing to login
- system time not synchronized periodically

Modeling Behavior

- What is normal activity? Must represent it
  - Periodic events
  - Sequence of events
  - Combination of events
  - Frequency of events
- Allows detection of missing events
- Allows verification of normal operation
- Disadvantages
  - Initial cost to model is high
  - Must maintain model over time

Modeling Topology

- What does our system look like?
  - What devices are there?
  - What services are there?
  - How do they depend on each other?
- Graph-based representation
- Helps determine source or “root cause” of event
  - e.g., is a service down because a network device failed?
- Often used for mapping networks

Correlating Events

- **Correlate**: assign a meaning to events
  - Pair: associate one event with another
  - Count: similar events occurring in time period
    - Threshold event: exceeds preset amount
    - Frequent event: amount per time period
  - Thread: combine related events
    - Sequence: events occur in order
    - Unordered: events are not related by time
  - Deduplicate: suppress subsequent same events
  - User-defined

Reason for Event Correlation

- **/var/log/messages**

Feb 14 19:31:10 gate2 pam_winbind[27607]: request failed: No such user, PAM error was User not known to the underlying authentication module (10), NT error was NT_STATUS_NO_SUCH_USER

Feb 14 19:31:10 gate2 sshd(pam_unix)[27607]: authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=c-24-19-144-115.hsd1.wa.comcast.net user=labadmin

Feb 14 19:31:14 gate2 pam_winbind[27607]: request failed: No such user, PAM error was User not known to the underlying authentication module (10), NT error was NT_STATUS_NO_SUCH_USER

Feb 14 19:31:18 gate2 pam_winbind[27607]: request failed: No such user, PAM error was User not known to the underlying authentication module (10), NT error was NT_STATUS_NO_SUCH_USER

Feb 14 19:31:22 gate2 sshd(pam_unix)[27607]: 5 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=c-24-19-144-115.hsd1.wa.comcast.net user=labadmin

Feb 14 19:31:22 gate2 sshd(pam_unix)[27607]: service(sshd) ignoring max retries; 6 > 3

- **/var/log/secure**

Feb 14 19:31:13 gate2 sshd[27607]: Failed password for labadmin from ::ffff:24.19.144.115 port 1876 ssh2

Feb 14 19:31:17 gate2 sshd[27607]: Failed password for labadmin from ::ffff:24.19.144.115 port 1876 ssh2

Feb 14 19:31:20 gate2 sshd[27607]: Failed password for labadmin from ::ffff:24.19.144.115 port 1876 ssh2

Correlating Events (cont.)

- How to correlate?
  - Formulate rule
    - Express condition-action pairs
    - Seem natural; can be readable and maintainable
  - Build statistical model
    - Related events have statistical similarities in attributes
      - Attributes are key parts of events
    - Use probabilities from prior events to relate current event
  - Develop codebook
    - Encode representative set of attributes or events
    - Closest match of current encoding to saved encodings
  - Build neural net (auto-associative)
    - Create clusters based on similar attributes
    - Clusters of events are correlated; non-clustered are interesting

Commercial Approaches

- According to Gartner (2006):
  - All: accept and process events; alert on critical events; take corrective action where possible
  - Often-employed Technologies
    - Network-centric approach, with auto-discovery
    - Automatic analysis of root cause
    - Help with defining/detecting abnormal events
    - Model and/or rule-based correlation
  - Frontrunners (usually expensive)
    - HP OpenView, IBM Tivoli, CA Unicenter (?), Microsoft Operations Manager
  - Specialized, upcoming or not as popular (some low-cost)
    - EMC Smarts, BMC Software, NetIQ, Quest Software, Nimsoft, Interlink Software, Argent Software, PerformanceIT, OpenService, TNT Software, Entuity, Rocket Software

Rule Based Correlation: SEC

- Simple Event Correlator, by Risto Vaarandi
  - Rule-based
  - Can process multiple input streams, static and dynamic
  - Can generate events, and save/refer to state
  - Written in Perl for portability and pattern-matching
  - Handles most event operations and allows scheduling
    - Match single event, match paired events, compress, count with thresholds and frequency
  - Fairly efficient
  - Used widely for IDS, fault detection, etc.
  - Free, with several good documents on how to use
    - From author and contributors

Reason for Event Correlation

- **/root/rules/login_failed.cfg**

```
# Sample input:
# /var/log/messages
# Feb 14 19:31:10 gate2 sshd(pam_unix)[27607]: authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
      rhost=c-24-19-144-115.hsd1.wa.comcast.net  user=labadmin
# /var/log/secure
# Feb 14 19:31:13 gate2 sshd[27607]: Failed password for labadmin from ::ffff:24.19.144.115 port 1876 ssh2
#

type=Pair
ptype=RegExp
pattern=\[(\d+)\]: authentication failure;.+? rhost=(\S+)\s+user=(\S+)
desc=authentication failure pid $1, user $3 from host $2
action=write - authentication failure, but no failed password for $3 from host $2
ptype2=RegExp
pattern2=\[(\d+)\]: Failed password for (\S+)
desc2=Failed password for $2
action2=write - Failed password for $2
window=30
```

- perl /usr/local/sbin/sec --conf=/root/rules/login_failed.cfg --input=/var/log/messages –input=/var/log/secure

Future Directions

- Already areas of research, but expect more investigation of and improvements in:
  - automatic detection of rules/patterns
  - integration and use of databases
  - integration of modeling and analysis
  - mining of event data
  - performance improvements
  - standardization of events

Conclusion

- Events are a necessary part of computing
- Handling events is labor-intensive and error-prone
- Many tools exist to assist system admins in:
  - filtering large numbers of events
  - determining the root cause of a problem
  - modeling events
  - correlating events
  - minimizing alerts
- By using these tools, you may be able to improve the availability and security of your systems

References

- http://www.loganalysis.com
- Spectrum: (now part of CA)
  - http://www.aprisma.com/literature/white-papers/wp0536.pdf
- Event correlation links:
  - http://wwwmnmteam.informatik.uni-muenchen.de/projects/evcorr/
- Gartner 2006: Event Correlation and Analysis
  - http://mediaproducts.gartner.com/reprints/computerassociates/139655.html
- Auto-association:
  - http://www.site.uottawa.ca/~nat/Papers/Dondo_Nat.pdf
- Statistical:
  - http://www.sdl.sri.com/papers/raid2001-pac/prob_corr.pdf
- SEC:
  - http://simple-evcorr.sourceforge.net