Departamento de ENGENHARIA INFORMÁTICA
Instituto Superior de Engenharia do Porto

Systems and Information Security SEGSI

Topic 6

Monitoring

Pinto Leite, Jorge (jpl@isep.ipp.pt)

Monitoring

- It is, unfortunately, more likely to have a reaction after a problem than to anticipate it
  - Anticipation is usually more difficult and more prone to human error
- However, it can be helped by a robust monitoring system
- Monitoring is a process of observing and tracking activities and progress
- It can be applied to every asset that is part of the infrastructure
  - Hardware
  - Software

Monitoring

- Since the introduction of *Simple Network Management Protocol* (SNMP) almost every hardware device state can be observed, thus allowing an intervention if needed and appropriate
  - Depending of the device, the interventions can be performed automatically, others requiring an human effort
  - By *automatically* it should be noted that an human intervention can be also required; automatism just provides a method of immediate troubleshooting
- Monitoring is more hardware-oriented than software-oriented, nevertheless some mix of hardware and software actions can be used
- The term *monitoring* can also be stated as *supervisioning*
- Talking about monitoring without the use of the word *evaluating* is naive

Logging vs Monitoring

- Logging consists of registering events
  - On servers
  - In databases
  - In networks
  - In assets, in general
- Monitoring consists of analyzing events
  - Real-time (online) or time-delayed (offline)
  - That can be contained in logs (analysis)
  - With the aim of
    - Detect faults or anomalies
    - Detect trends

Logging (record of occurrences)

- In the context of security engineering
  - Why the occurrence log?
    - Registering users' actions later allows them to be held responsible for their actions (non-repudiation)
      - To do this, it is necessary to guarantee the integrity of the logs
      - It is also necessary to provide logging applications with security mechanisms, in order to exclude the possibility of someone impersonating another user
      - Can be used as evidence in legal proceedings
  - The knowledge that certain actions or events are recorded acts as a deterrent to possible offenders

Logging (record of occurrences)

- In the context of security
  - Why the occurrence log?
    - It is possible, using analysis and monitoring tools, to study behavior patterns of users, objects and networks
      - Behaviors that deviate from the pattern can signal possible security breaches
    - To assess the effectiveness of security features
    - Lets you discover repeated attempts to breach security
    - For attack analysis

Logging (record of occurrences)

- Outside the security context
  - Why the occurrence log?
    - Performance analysis
    - Accounting records
    - Data management (versions, transactions, etc.)
    - Compliance with legal requirements
    - Others...

Events to register in a network

- User account management events
  - Adding/removing accounts
  - Changes to security attributes
    - Access levels, login intervals, etc
  - Account suspensions/reactivations
  - Administrative password reset
  - Sudden increase in resource spent
  - Accesses in “abnormal” time periods
  - Etc.

Events to register in a network

- Access control events
  - Logins and logoffs (successful and unsuccessful)
  - Account access denied
    - Invalid passwords
    - Inactive sessions
    - Access using unauthorized interfaces
    - Attempts to login during unauthorized periods
    - Violation of the limit of simultaneous sessions
  - Changing passwords
    - Change frequency
    - Time of change

Events to register in a network

- Changing settings
  - Changing the configuration of critical functions for certain important applications
    - Examples
      - Interest rates
      - Prices
  - Changing system parameters
    - Examples
      - Password length
      - Maximum number of connections per user

Events to register in a network

- Attempts to access applications and system resources
  - Cryptographic key changes
  - Starting/stopping of
    - Services/process/applications
  - Unexpected application disruptions
  - Attempts to fail to connect to databases
  - Attempt to change critical information on operating systems (registry, LSB, etc.)

Events to register in a network

- Attempts to access applications and system resources...
  - Logins/logoffs due to system maintenance
  - Failures to verify the integrity of
    - Application data
    - Executables
    - Logs
  - Access to applications/resources without having the necessary licenses
- User-entered commands
  - su ; rm –fR * ; fdisk ; …
  - format C: ; …

Events to register in a network

- Performance of the System / Network
  - Unusual indicators may be one of the first indicators of attacks or may suggest the imminence of an attack
    - Examples
      - Network
      - Packets of a certain type
      - Connections from the same remote IP
    - System
      - CPU load
      - Processes
      - Virtual memory

Events to register in a network

- Network traffic
  - Traffic arriving at the network (from an external network)
    - By default all traffic should be registered
  - Traffic leaving the network (for an external network)
    - Identify machines and services that are not initially expected to send packets out of the network
    - Log traffic for protocols considered unsafe and important to the company
    - Monitor source spoofing of IP addresses
    - Etc.

Level of detail

- Compromise between adequate level of detail and system performance
- For each event it is important to register
  - The event ID and its type
  - Date (timestamp)
  - Error message (if applicable)
  - Success or failure of the event
  - IP address of client (if applicable)
  - ID of user that originated/provoked the event
  - Accessed resources
  - Concrete actions

Log management

- Collect and aggregate logs
  - In order to allow
    - Log analysis
    - Set alerts
    - Archive logs
  - They must be collected and transported in a safe and reliable way
- Log volume can reach terabytes
- A solution is needed that
  - Automatically analyze logs
  - Automatically trigger alerts
  - Produce reports
- Dramatic reduction of data that must be analyzed manually is vital

Log management

- Log analysis
  - Pattern detection (grep, awk, perl)
  - Correlation of entries
  - *Root Cause Analysis*
    - The first time that an unknown or unexpected occurrence happened
    - It usually allows detection of *what went wrong that allowed the situation*
- Set alerts
  - Moving to “illegal” states
  - Significant deviations from “normal” standards
  - Exceeding thresholds
  - Etc.

Log management

- Archive Logs
  - In the USA the way they are archived is certified
  - Archived certified logs can serve as evidence in court
  - Generate two copies of logs (good practice)
    - One for monitoring
    - Another for security
  - Use secure encryption/compression
  - Automate (limit human intervention)
    - Reduce operating costs
    - Increase process reliability
    - Decrease the probability of tampering of logs
    - Possibility of faster reaction (in simple events)
    - Etc.

Security considerations

- Logs must not be stored on the system itself
  - If the system is compromised, the logs can be removed or tampered
- It is crucial to reinforce security in the system(s) where the logs are stored
- Restricted access to logs
  - Define to what, who and why
  - Secure encryption, integrity control and the need for strong authentication
- Never record security credentials along with logs (why?)
  - Passwords, PINs, encryption keys, etc
- The logs must be archived periodically
  - Rotating logs
  - Use remote storage locations
- Integrity control mechanisms must exist
  - Using cryptography is highly recommended

Logging tools

- Types of tools
  - Centralized
    - Each event is sent to a dedicated server
    - It is necessary to protect communications and access to the dedicated server
    - Server(s) timing is important
    - It is a “central point of failure”
    - Management is easier
  - Distributed
    - Each machine contains its share of the logs
    - There have to be synchronization mechanisms
    - Management is more difficult

Logging tools

- Storage mechanism
  - Text files
  - CSV
  - Binary format
  - In databases
- With / without graphical interface
  - Almost all of them support console interface
  - The graphical interface allows some advantageous forms of critical or consolidated visualization

Logging tools

- System Logs
  - Syslog / Syslog-ng -> Unix/Linux
  - Eventlog, perfmon -> Windows
  - SNMP traps -> Generic log mechanisms
- Logging applications
  - Logcheck (Unix/Linux)
  - Logwatch (Unix/Linux)
  - Logrotate (Unix/Linux)
- Network Logging
- Wireshark, tcpdump, iptables, arpwatch, etc.

Syslog

- Allows message classification by level and by area
  - Levels
    - Warning
    - Error
    - Emergency
  - Area
    - Printing
    - Email
    - Network

Syslog – message levels

- LOG_EMERG
  - Panic condition: the message is usually broadcast to all users
- LOG_ALERT
  - A situation that must be corrected immediately (example: corrupted database)
- LOG_CRIT
  - Critical situations (example: disk errors)
- LOG_ERR
  - Errors
- LOG_WARNING
  - Warning messages
- LOG_NOTICE
  - Messages that, not being errors, should be analyzed with special attention
- LOG_INFO
  - Informational messages
- LOG_DEBUG
  - Messages that contain debugging information

Syslog – message levels

- LOG_KERN
  - Messages generated by kernel
- LOG_DAEMON
  - Messages related to system services (examples: ftpd and sshd)
- LOG_AUTH
  - Messages related with authentication (examples: login, su, sshd)
- LOG_USER
  - Messages generated by user process (by default)
- LOG_MAIL
  - Messages related to the email system
- LOG_LPR
  - Printing-related messages (examples: lpr, cups, lpd)
- LOG_LOCAL0 up to LOG_LOCAL7
  - Reserved for local use
- Others
  - Created by the administrator

Syslog – syslog.conf

- *Daemon* syslogd
- Configuration in /etc/syslog.conf

  <area>.<level> <destination>

  - It is possible to use wildcards (*)
  - The “none” level is used to eliminate area (example: area.none)
  - It is possible to combine several types and areas (with “;”)
  - Examples

```
mail.*                  /var/log/maillog
*.info                  /var/log/messages
*.emerg                 *
kern.*                  console
*.*                     @logmachine
*.info;mail.none        /var/log/messages
*.*                @@server1.example.net:10514
```

Append to file

Sends info to all logged users

Displays on the console

Sends to logserver

Use TCP instead of UDP

Sends to port 10514 of the server

Syslog – example

Jun 7 23:44:19 tux sshd(pam_unix)[7529]: session opened for user pedro by (uid=500)

Jun 7 23:44:26 tux su(pam_unix)[7572]: session opened for user root by pedro (uid=500)

Jun 8 00:00:00 tux nagios: LOG ROTATION: DAILY

Jun 8 00:00:34 tux su(pam_unix)[7572]: session closed for user root Jun 8 00:00:37 tux sshd(pam_unix)[7529]: session closed for user pedro

Jun 8 01:43:38 tux nagios: Auto-save of retention data completed successfully.

Jun 8 02:00:04 tux nagios: Warning: A system time change of 1 seconds (backwards in time) has been detected. Compensating...

Jun 8 04:43:37 tux nagios: Auto-save of retention data completed successfully.

Jun 8 05:12:21 tux dhcpd: DHCPREQUEST for 192.168.100.7 from 00:11:d8:a5:6b:db via eth0

Jun 8 05:12:21 tux dhcpd: DHCPACK on 192.168.100.7 to 00:11:d8:a5:6b:db via eth0 Jun 8 05:12:24 tux dhcpd: DHCPINFORM from 192.168.100.7 via eth0: not authoritative

for subnet 192.168.100.0

Jun 8 05:13:52 tux last message repeated 2 times

Jun 8 05:13:55 tux dhcpd: DHCPINFORM from 192.168.100.7 via eth0: not authoritative for subnet 192.168.100.0

Jun 8 05:32:18 tux smbd[9446]: [2007/06/08 05:32:18, 0]

lib/util_sock.c:read_socket_data(342)

Jun 8 05:32:18 tux smbd[9446]: read_socket_data: recv failure for 4. Error = No route to host

Jun 8 05:43:37 tux nagios: Auto-save of retention data completed successfully.

Monitoring

- It describes all processes to measure and evaluate specific data under use of technical tools
- By using monitoring software (like Nagios, Icinga, among others) security (and, obviously, business) is being able to collect and evaluate specific data
- The question here is what, how, and for why that data is being collected and evaluated
- In addition of hardware monitoring (CPU temperature, RAM in use, disk occupation, etc.), it can / should get inputs from available logs
- In foreground Key Performance Indicators (KPI) must have been defined

Monitoring

- KPI's are quantifiable measurements that can help gauge a company's or an organization's progress towards its strategic objectives
  - Which leads, by its turn, to BCMS
- Some usual monitoring types are
  - Application performance
  - Business transaction
  - System

Monitoring

- Security and Incident Management (SIEM) is a security solution that can provide an overall view of security threats
- It collect logs from security information management (SIM) and security event management (SEM) thus allowing their correlation and to implement actions (at least, an alert) when a problem happens or is likely to happen

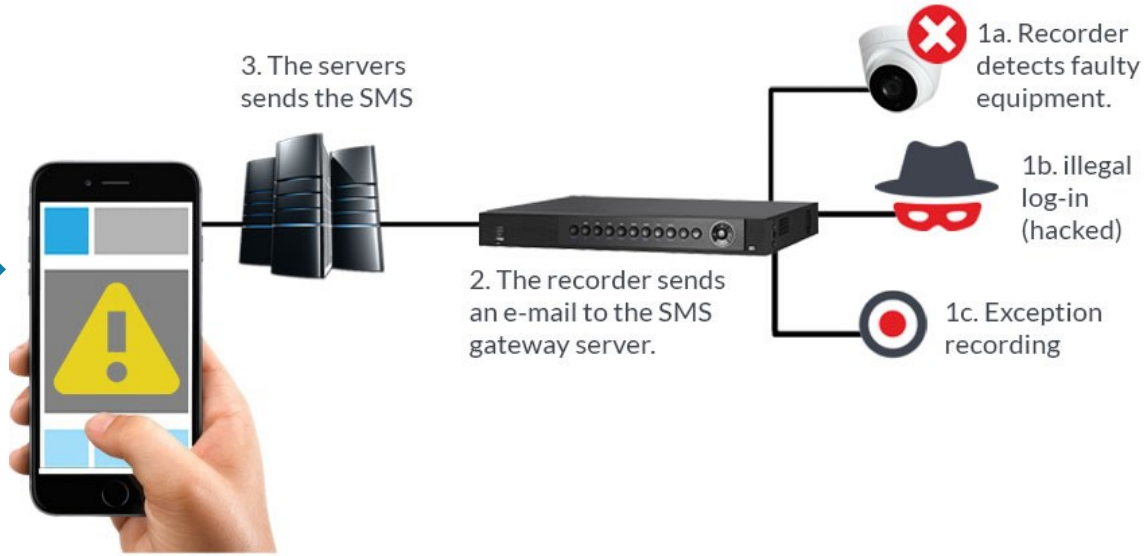
Monitoring

- Application performance monitoring
  - Its goal is to check functionality of applications and programs to improve user experience
- Business transaction monitoring
  - Its goal is to supervise business processes
  - By evaluating those outputs, an optimization of the processes that are directly connected to business transaction can be considered and implemented
- System monitoring
  - Its goal is to review the performance of the system
  - *System* should be read as *infrastructure*, not as a single computer or similar

Monitoring

- Monitoring by itself is useless
- Appropriate mechanisms should be implemented to allow an action to be performed timely

Source: intec



Source: innotec

Monitoring

- Monitoring calls the alarmist to advise whoever is needed to carry out the tasks
- This does not mean that monitoring should only call alarmist when a problem is detected
- Instead, it can be configured to launch an alarm when controls reports environments that *can* probably result on critical situations – including asset failure