

Ανίχνευση εισβολών

Νικόλαος Ε. Κολοκοτρώνης
Επίκουρος Καθηγητής

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Πελοποννήσου

Email: nkolok@uop.gr

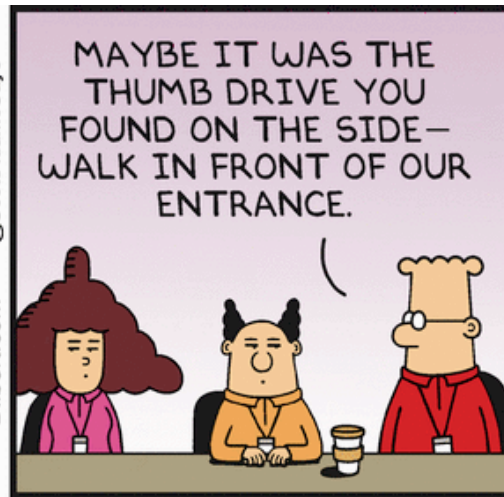
Web: <http://www.uop.gr/~nkolok/>

ΑΣΦΑΛΕΙΑ ΣΥΣΤΗΜΑΤΩΝ

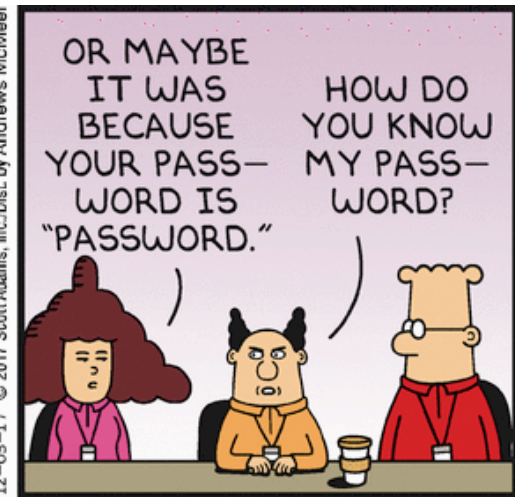
Περιεχόμενα



Dilbert.com @ScottAdamsSays



12-05-17 © 2017 Scott Adams, Inc./Dist. by Andrews McMeel



Περιεχόμενα

- Intruders
 - masquerader
 - misfeasor
 - clandestine user
- Intruder behavior patterns
 - hacker
 - criminal enterprise
 - internal threat
- Intrusion detection systems
 - host-based
 - network-based
- Detection techniques
 - anomaly detection
 - signature detection
- Honeypot
- SNORT / others

Intruders

- Key threats to security is the use of hacking by an intruder
 - often referred to as a hacker or cracker
 - small number of very large dataset compromises by insiders
 - most security breaches are due to by outsiders
- There is a general increase in malicious hacking activity
 - attacks targeted at users in organizations and their IT systems
 - targeted attacks are designed to bypass perimeter defenses like firewalls and network-based IDSs
- Need to use defense-in-depth strategies

Intruders

Cyber criminals

- individuals or members of a group
- financial rewards or goals
- activities include identity theft, theft of financial credentials, corporate espionage, data theft, or data ransoming

Hacktivists

- individuals (usually insiders) or members of a group (outsiders)
- e.g. *Anonymous*
- social or political motivations
- their skill level is often quite low
- activities include defacement of website, DoS attacks, or data theft/distribution

State-sponsored organizations

- groups of hackers sponsored by governments
- e.g. Advanced Persistent Threats (APTs)
- activities include conduct of espionage or sabotage

Others

- classic hackers or crackers
- motivations like peer-group reputation or technical challenge
- Mainly responsible for discovering new forms of vulnerabilities

Intrusion examples

- Remote root compromise
- Web server defacement
- Guessing and cracking passwords
- Copying databases containing credit card numbers
- Viewing sensitive data without authorization
- Running a packet sniffer
- Distributing pirated software
- Impersonating an executive to get information

Hackers

- Motivated by thrill of access and/or status
 - hacker's status is determined by level of competence
- Benign intruders slow performance and consume the resources of legitimate users
- IDSs and IPSs are designed to help counter hacker threats
 - can restrict remote logons to specific IP addresses
 - can use VPN technology
- Intruder problem led to establishment of CERTs

CERTs or CSIRTs

- Computer emergency response teams
- Cooperative ventures collecting information about system vulnerabilities and disseminate it to security admins, e.g.
 - <http://www.kb.cert.org/vuls>
 - <http://cve.mitre.org>
- Hackers also routinely read CERT reports
- It is important for system administrators to quickly insert all software patches to discovered vulnerabilities

Hackers' behavioral patterns

1

- select target using IP lookup tools like NSLookup, Dig, etc.

2

- map network for accessible services using tools such as NMAP

3

- identify potentially vulnerable services

4

- brute force (guess) pcAnywhere password

5

- install remote administration tool called DameWare

6

- wait for administrator to log on and capture his password

7

- use that password to access remainder of network

Cyber-criminals

- Organized groups of hackers now a threat
 - corporation / government / loosely affiliated gangs
 - typically young
 - often Eastern European, Russian, or southeast Asian hackers
 - meet in underground forums
 - common target is credit card files on e-commerce servers
- Criminal hackers usually have specific targets
 - once penetrated act quickly and get out
- Sensitive data should always rest in encrypted form

Cyber-criminals' patterns

Act quickly and precisely to make their activities harder to detect



Exploit perimeter via vulnerable ports



Use Trojan horses (hidden software) to leave back doors for re-entry



Use sniffers to capture passwords



Do not stick around until noticed

Insider attacks

- Among most difficult to detect and prevent
- Employees have access and systems knowledge
- May be motivated by revenge/entitlement
 - employment was terminated
 - taking customer data when moving to a competitor
- IDS / IPS can be useful but also need countermeasures
 - least privilege enforcement, logs monitoring, strong authentication, process termination

Insider attacks: countermeasures

- Enforce least privilege, only allowing access to resources by employees that need to do their job
- Set logs to see what users access and what commands they are entering
- Protect sensitive resources with strong authentication
- Upon termination, delete employee's computer and network access
- Upon termination, make a mirror image of employee's HDD before reissuing it (can be used as evidence)

Intrusion techniques

- Objective of the intruder is to gain access to a system or to increase the range of privileges accessible on a system
- Most attacks use system/software vulnerabilities allowing a user to execute code opening a backdoor into the system
- Protecting a password file:

One-way functions

- The system stores only the value of a function based on the user's password

Access control

- Access to the password file is limited to one or a very few accounts

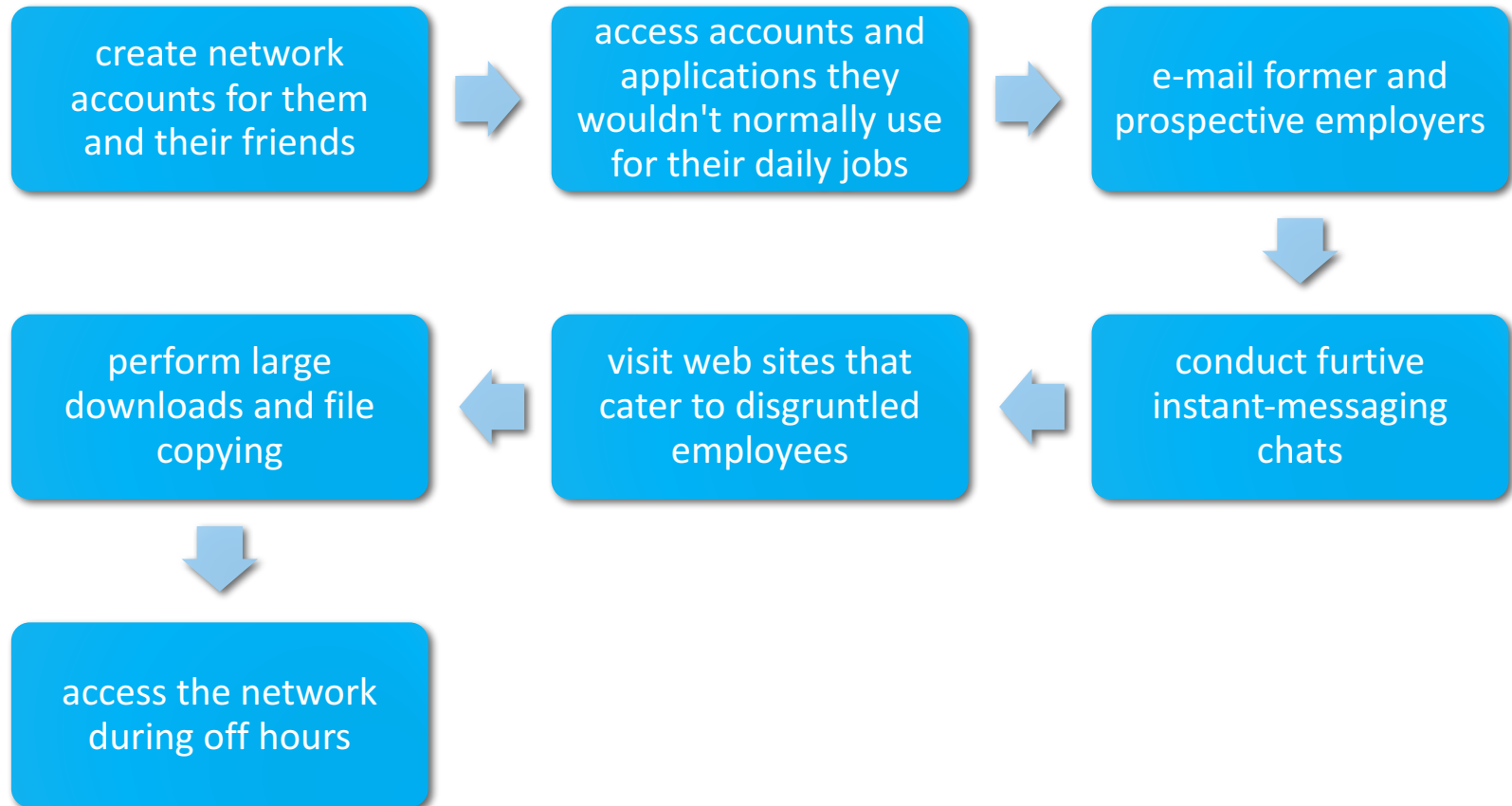
Password guessing

- Try default passwords used with standard accounts that are shipped with the system
 - Many administrators do not bother to change these defaults
 - This is the source of so many attacks
- Exhaustively try all short passwords
 - e.g. those of 1-4 characters
- Try words in the system's online dictionary or a list of likely passwords
 - readily available on hacker bulletin boards

Password guessing

- Collect adequate information about the users, e.g.
 - full names, names of spouse and children
 - office pictures, office books, hobby-related
- Try users' phone numbers, social security numbers, and room numbers
- Try all legitimate license plate numbers for this state
- Use a Trojan horse to bypass restrictions on access
- Tap the line between a remote user and the host system

Threat behavioral patterns



Relevant RFC 2828 definitions

- Security intrusion

- A security event, or a combination of multiple security events, that constitutes a security incident in which an intruder gains (or attempts to gain) access to a system or resource without having authorization to do so

- Intrusion detection

- A security service that monitors and analyzes system events for the purpose of finding and providing real-time (or near real-time) warning of attempts to access system resources in an unauthorized manner

Intrusion detection systems

comprises three logical components

- Sensors – collect data
- Analyzers – determine if intrusion has occurred
- User interface – view output or control system behavior

■ Host-based IDS

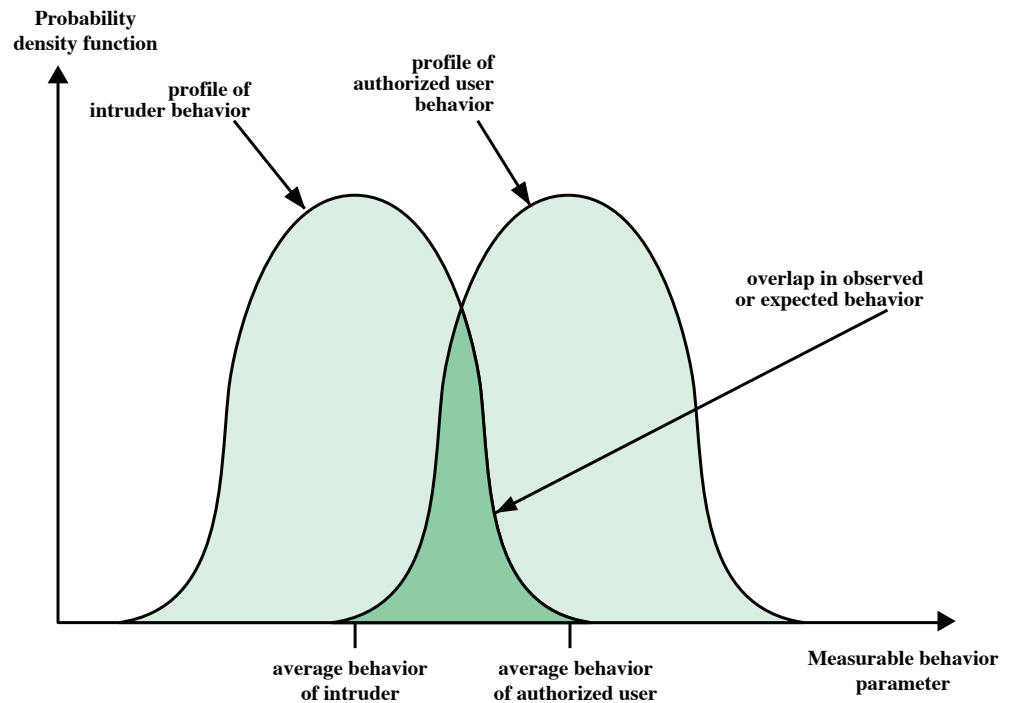
- monitors the characteristics of a single host for suspicious activity

■ Network-based IDS

- monitors network traffic and analyzes network, transport, and application protocols to identify suspicious activity

IDS principles

- Assume intruder behavior differs from the legitimate users' behavior
- Overlap in the behaviors causes problems
 - false positives
 - false negatives



IDS requirements

run continually

be fault tolerant

resist subversion

impose a minimal
overhead on system

configured
according to system
security policies

adapt to changes in
systems and users

scale to monitor
large numbers of
systems

provide graceful
degradation of
service

allow dynamic
reconfiguration

Host-based IDS

- Adds a specialized layer of security software to vulnerable or sensitive systems
- Monitors activity to detect suspicious behavior
 - primary purpose is to detect intrusions, log suspicious events, and send alerts
 - can detect both external and internal intrusions

Host-based IDS: approaches

ANOMALY DETECTION

- Threshold detection
 - involves counting the number of occurrences of a specific event type in a time interval
- Profile-based
 - profile of a user's activity is developed and used to detect changes in the behavior

SIGNATURE DETECTION

- Involves an attempt to define a set of rules or attack patterns that can be used to decide that a given behavior is that of an intruder
- Often referred to as *rule-based* detection

Statistical anomaly detection

■ Threshold detection

- Involves counting the number of occurrences of a specific event type in a time interval
- If the count exceeds a reasonable number that one might expect to occur, then intrusion is assumed
- An ineffective detector for even moderately advanced attacks

■ Profile-based

- Focuses on characterizing the past behavior of individual users or (user groups) and then detecting significant deviations
- A profile may consist of a parameter set, so that deviation on a single value will not suffice to signal an alert

Audit records

Native audit records

- Multiuser OS include accounting software that collects info on user activity
- Advantage: no additional collection software is needed
- Disadvantage: records may not contain the needed information or in a convenient form

Detection-specific audit record

- Collection facility that generates records only with info required by the IDS
- Advantage: it could be made vendor and platform / OS independent
- Disadvantage: the extra overhead of having, in effect, two accounting packages running on a machine

IDS measures: login activity

Measure	Model	Intrusion detection type
Login frequency by day and time	Mean and standard deviation	Intruders may be likely to log in during off-hours.
Frequency of login at different locations	Mean and standard deviation	Intruders may log in from a location that a particular user rarely or never uses.
Time since last login	Operational	Break-in on a "dead" account.
Elapsed time per session	Mean and standard deviation	Significant deviations might indicate masquerader.
Quantity of output to location	Mean and standard deviation	Excessive amounts of data transmitted to remote locations could signify leakage of sensitive data.
Session resource utilization	Mean and standard deviation	Unusual processor or I/O levels could signal an intruder.
Password failures at login	Operational	Attempted break-in by password guessing.
Failures to login from specified terminals	Operational	Attempted break-in.

IDS measures: program activity

Measure	Model	Intrusion detection type
Execution frequency	Mean and standard deviation	May detect intruders, who are likely to use different commands, or a successful penetration by a legitimate user, who has gained access to privileged commands. An abnormal value might suggest injection of a virus or Trojan horse, which performs side-effects that increase I/O or processor utilization. May detect penetration attempt by individual user who seeks higher privileges.
Program resource utilization	Mean and standard deviation	
Execution denials	Operational model	

IDS measures: file access activity

Measure	Model	Intrusion detection type
Read, write, create, delete frequency	Mean and standard deviation	Abnormalities for read and write access for individual users may signify masquerading or browsing.
Records read, written	Mean and standard deviation	Abnormality could signify an attempt to obtain sensitive data by inference and aggregation.
Failure count for read, write, create, delete	Operational	May detect users who persistently attempt to access unauthorized files.

Signature detection

- Detect intrusion by observing events in the system
 - Apply a set of rules that leads to a decision regarding whether a given pattern of activity is or is not suspicious
- Signature anomaly detection
 - Similar in terms of its strengths to statistical anomaly detection
 - Historical audit records are analyzed to identify usage patterns and to automatically generate rules that describe those patterns
 - Current behavior is monitored, and each transaction is matched against the set of rules to see if it conforms to observed patterns
 - Is effective if a rather large database of rules is available

Signature detection

- Signature penetration identification

- The rules used are specific to the machine and operating system
- The most fruitful approach to developing such rules is to analyze attack tools and scripts collected on the Internet
- Rules generated by security personnel could also be added

- USTAT

- A model independent of specific audit records
- Deals with generic actions, not detailed as those recorded in UNIX
- Implemented on SunOS providing audit records on 239 events

USTAT vs. SunOS

Based on actions and
event types

USTAT Action	SunOS Event Type
Read	open_r, open_rc, open_rtc, open_rwc, open_rwtc, open_rt, open_rw, open_rwt
Write	truncate, ftruncate, creat, open_rtc, open_rwc, open_rwtc, open_rt, open_rw, open_rwt, open_w, open_wt, open_wc, open_wct
Create	mkdir, creat, open_rc, open_rtc, open_rwc, open_rwtc, open_wc, open_wtc, mknod
Delete	rmdir, unlink
Execute	exec, execve
Exit	exit
Modify_Owner	chown, fchown
Modify_Perm	chmod, fchmod
Rename	rename
Hardlink	link

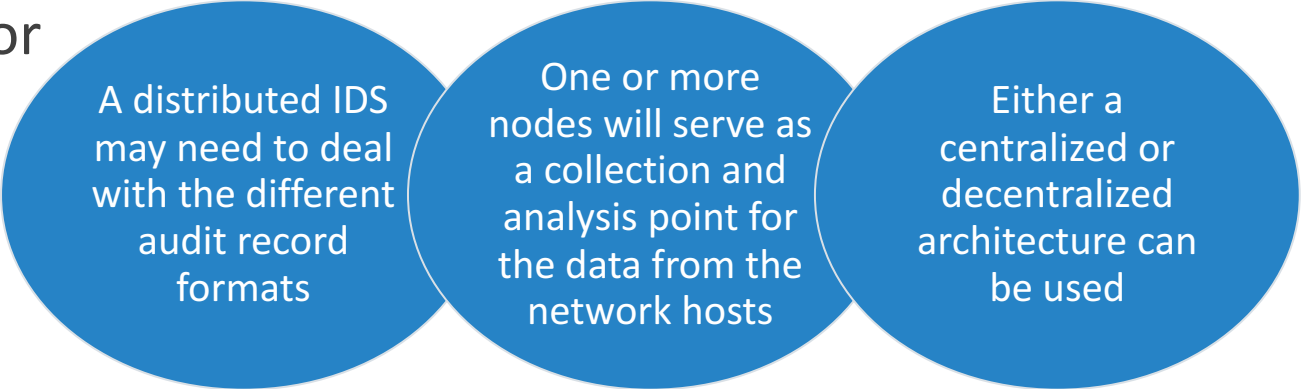
Base-rate fallacy

- To be of practical use, an IDS should be highly accurate, keeping the false alarm rate at an acceptable level
 - If only a modest percentage of actual intrusions are detected, the system provides a false sense of security
 - If the system frequently triggers an alert when there is no intrusion, most of the time will be wasted in analyzing false alarms
- In principle, it is difficult to meet the standard of high rate of detections with a low rate of false alarms
 - If the actual intrusions is low compared to the number of legit uses, false alarm rate will be high unless the test is very discriminating

Distributed intrusion detection

- Initially focused on single-system stand-alone facilities
 - The typical organization, however, needs to defend a distributed collection of hosts supported by a LAN or internetwork
 - A more effective defense can be achieved by coordination and cooperation among intrusion detection systems across the network

- The major design issues for IDS:

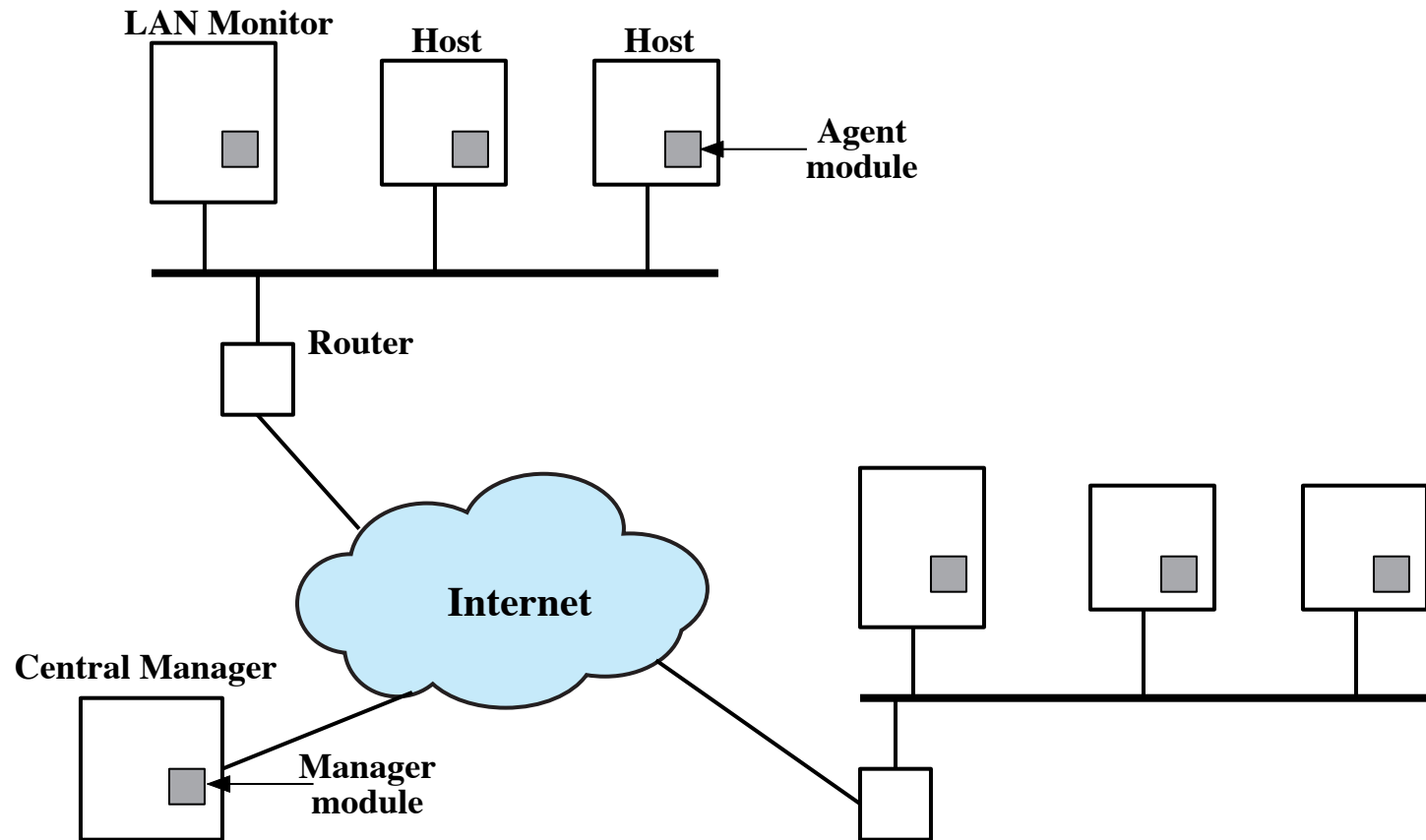


A distributed IDS may need to deal with the different audit record formats

One or more nodes will serve as a collection and analysis point for the data from the network hosts

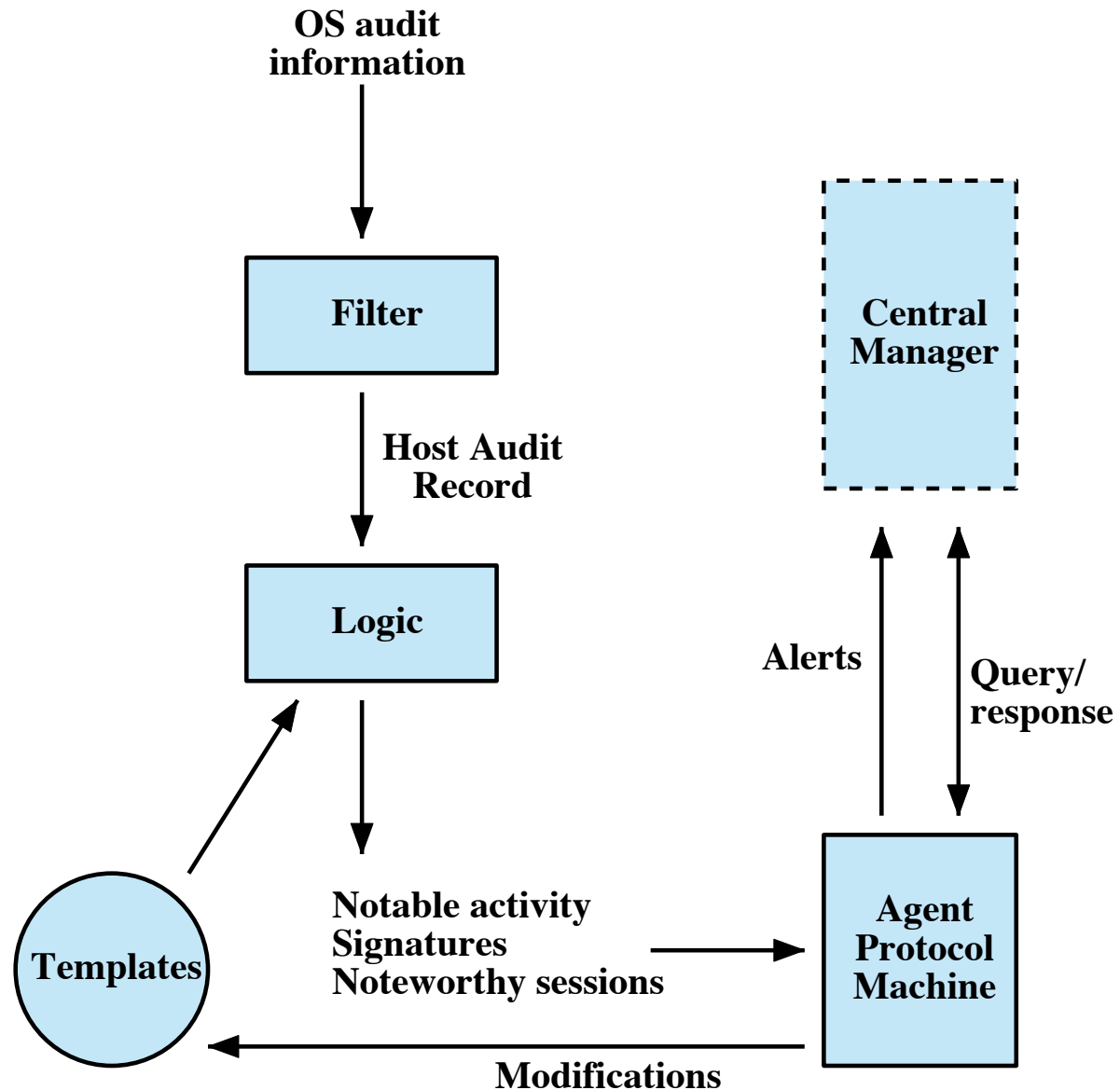
Either a centralized or decentralized architecture can be used

Distributed IDS architecture



Distributed host-based IDS

The agent's architecture



Network-based IDS (NIDS)

monitors traffic at selected points on a network

examines traffic packet by packet in real or close to real time

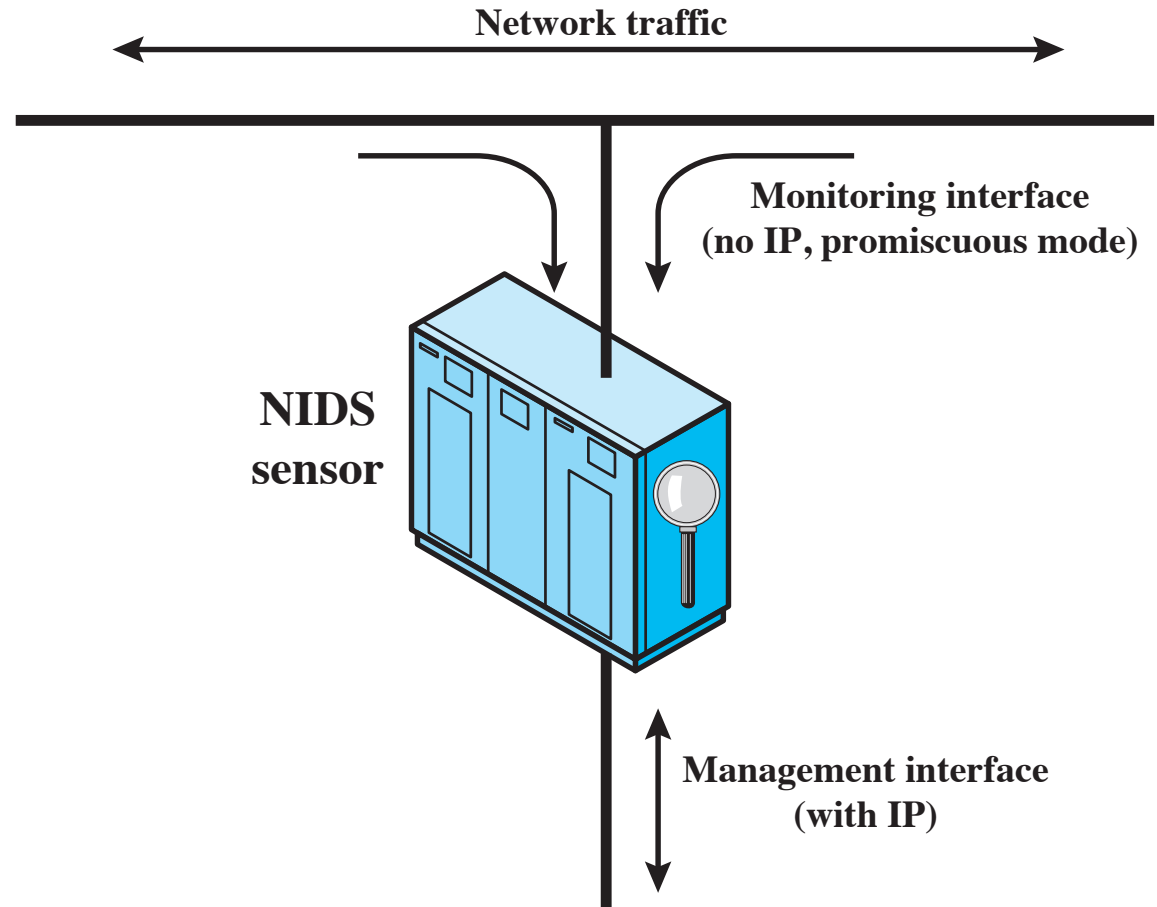
may examine network, transport, and/or application-level protocol activity

comprised of a number of sensors, one or more servers for NIDS management functions, and one or more management consoles for the human interface

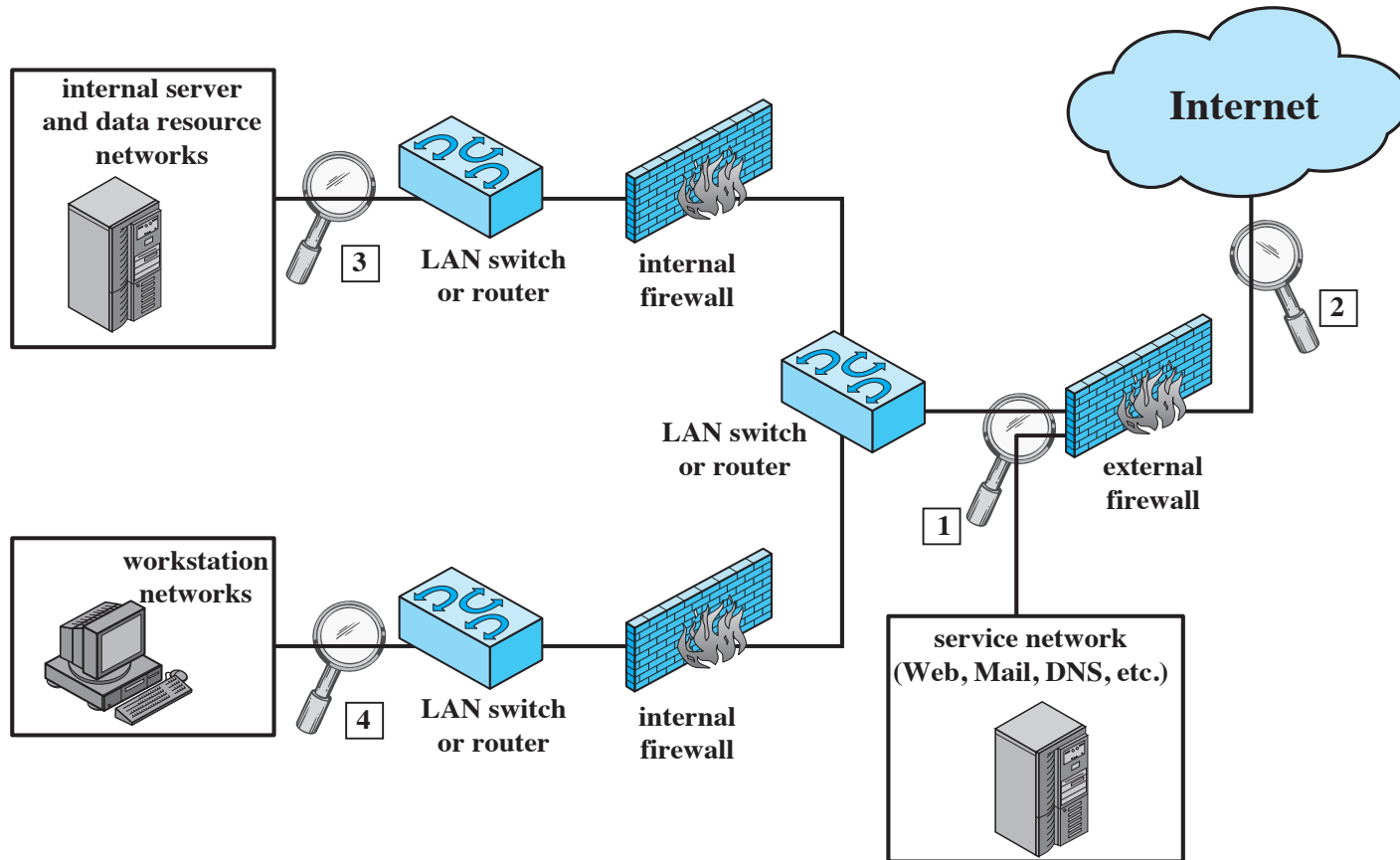
analysis of traffic patterns may be done at the sensor, the management server or a combination of the two

Passive NIDS sensor deployment

- **Inline sensor**
 - inserted into a network segment so that the traffic that it is monitoring must pass through the sensor
- **Passive sensors**
 - monitors a copy of network traffic



Example NIDS sensor deployment

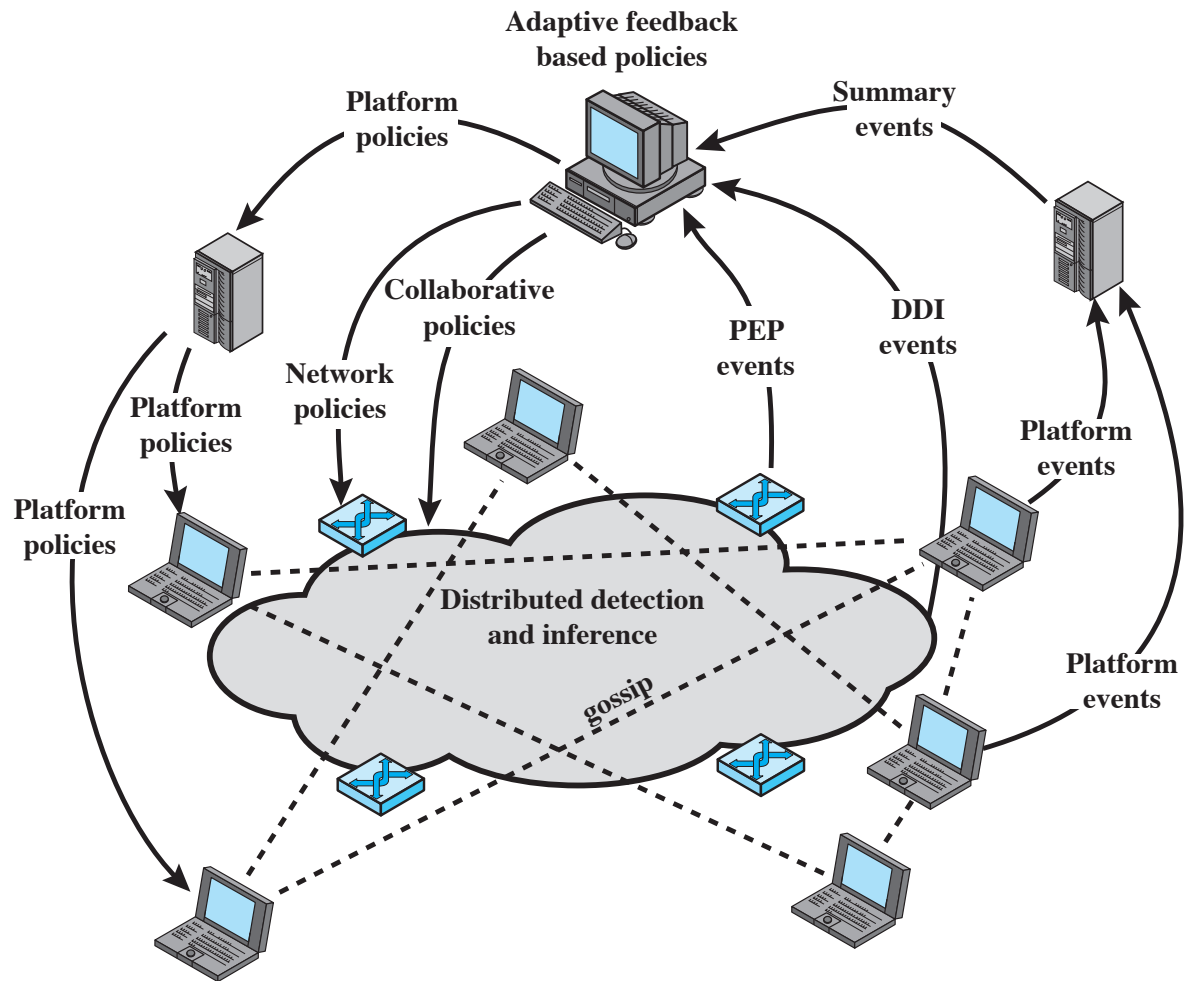


Detection techniques application

- Signature detection
 - at application, transport, network layers; unexpected application services, policy violations
- Anomaly detection
 - denial of service attacks, scanning, worms, etc.
- When a sensor detects a potential violation it sends an alert and logs information related to the event
 - used by analysis module to refine IDS parameters and algorithms
 - security admins use this info to design prevention techniques

Enterprise security system

Overall architecture

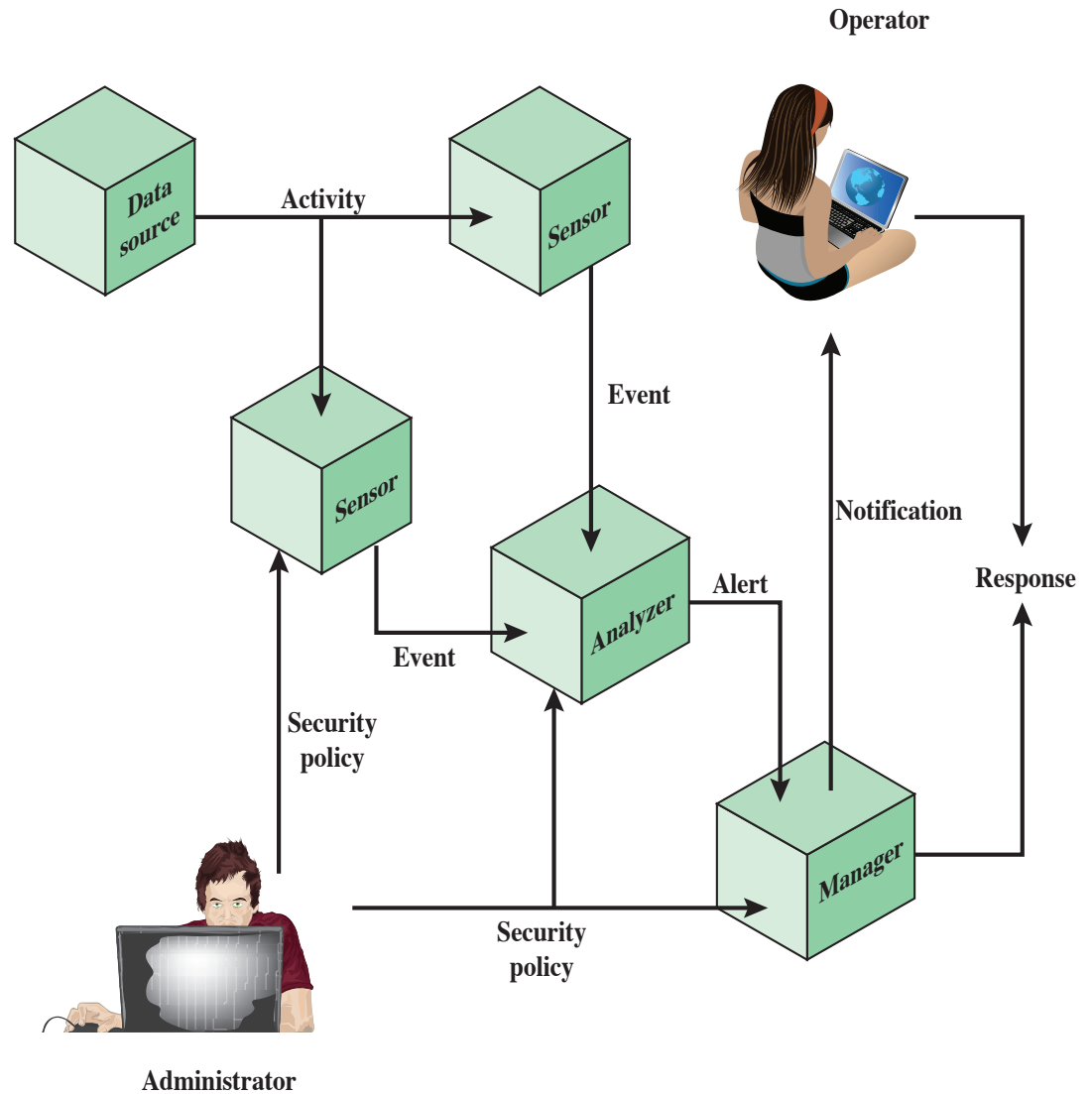


PEP = policy enforcement point
DDI = distributed detection and inference

Intrusion detection exchange

- Standards are needed to support interoperability
 - To facilitate the development of distributed IDS
 - Across a wide range of platforms and environments
- IETF intrusion detection WG
 - Purpose = define data formats and exchange procedures for sharing information related to intrusion detection with response systems
 - Have issued the following RFCs
 - ▶ Intrusion Detection Message Exchange Format (RFC 4765)
 - ▶ Intrusion Detection Message Exchange Requirements (RFC 4766)
 - ▶ Intrusion Detection Exchange Protocol (RFC 4767)

Intrusion detection message exchange

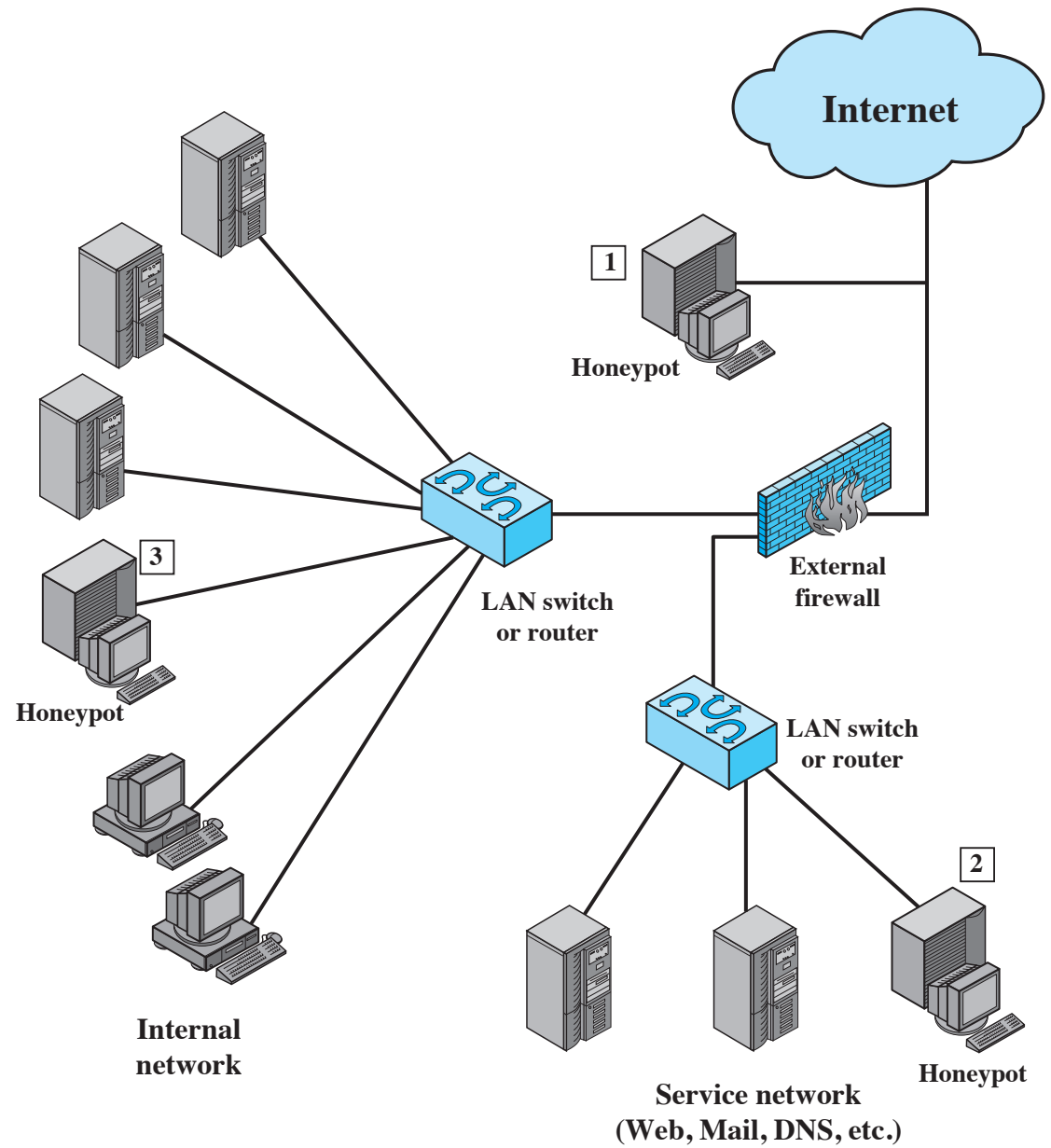


Honeypot

- Decoy systems designed to
 - lure a potential attacker away from critical systems
 - collect information about the attacker's activity
 - encourage the attacker to stay on the system long enough for administrators to respond
- Has fabricated information legit users wouldn't access
- Resource that has no production value
 - incoming communication is likely a probe, scan, or attack
 - outbound communication suggests that the system has probably been compromised
- Once hackers are in the network, admins observe behavior to pick defenses

Honeypot deployment

An example



SNORT

- The most popular (lightweight) NIDS
 - real-time packet capture and rule analysis
 - easily deployed on nodes and configured
 - needs for low memory and processor time
- A de-facto standard IDS in the practical security community
 - More than 3M downloads
 - More than 200K users

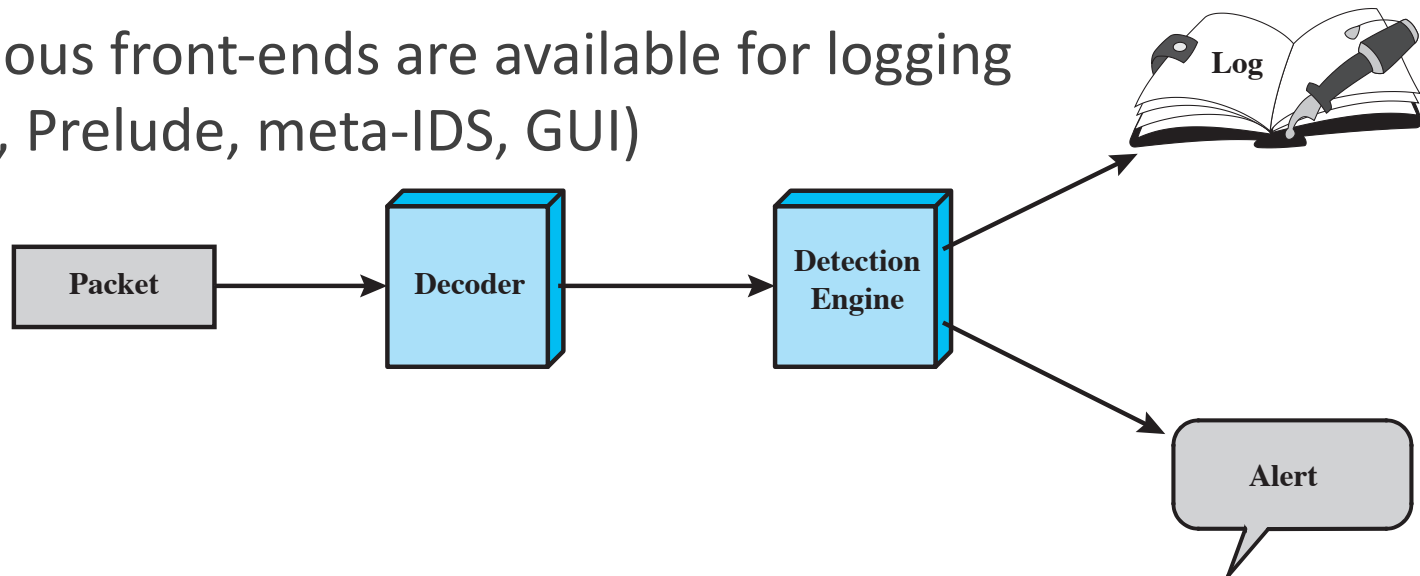


SNORT

- Είναι open source και χρησιμοποιεί
 - MySQL DB για την αποθήκευση των δεδομένων που καταγράφει
 - Apache web server και μια Analysis Control for Intrusion Database (ACID) για την επεξεργασία των δεδομένων
- Έχει δικούς του κανόνες ελέγχου εισβολών, αλλά δίνει τη δυνατότητα στο διαχειριστή να γράψει τους δικούς του
 - Αποθηκευμένα σε text files ανά κατηγορία
- Είναι διαθέσιμο για
 - Windows, Linux, Solaris, HP-UX, AIX, IRIX , MacOS, OpenBSD, ...

SNORT architecture

- The Decoder includes a packet sniffer and a preprocessor
 - Packet sniffer interacts directly with a network card using libpcap
 - Preprocessing, detection and alert components are plugins
- Various front-ends are available for logging (DB, Prelude, meta-IDS, GUI)



SNORT plug-ins

- Preprocessor

- Packets are examined/manipulated before being handed to the detection engine

- Detection

- Perform single, simple tests on a single aspect/field of the packet

- Output

- Report results from the other plug-ins

Preprocessor configurations

- Preprocessor configuration in “snort.conf”
 - frag2 detects packet fragmentation
 - stream4 self protection against Snot and Slick
 - http_inspect web traffic
 - rpc_decode RPC traffic
 - flow_portscan statistical details
 - sfportscan detect port scanning activities
 - perfmonitor Self assessment

Detection engine

- Rules form “signatures”
- Modular detection elements are combined to form these signatures
- Wide range of detection capabilities
 - Stealth scans, OS fingerprinting, buffer overflows, back doors, CGI exploits, etc.
- Rules system is very flexible, and creation of new rules is relatively simple

SNORT rules

- Uses a simple, flexible rule definition language
 - Rules consist of a fixed header and some options

Action	Description
alert	Generate an alert using the selected alert method, and then log the packet.
log	Log the packet.
pass	Ignore the packet.
activate	Alert and then turn on another dynamic rule.
dynamic	Remain idle until activated by an activate rule , then act as a log rule.
drop	Make iptables drop the packet and log the packet.
reject	Make iptables drop the packet, log it, and then send a TCP reset if the protocol is TCP or an ICMP port unreachable message if the protocol is UDP.
sdrop	Make iptables drop the packet but does not log it.

SNORT rules

Examples of rule options

meta-data	
msg	Defines the message to be sent when a packet generates an event.
reference	Defines a link to an external attack identification system, which provides additional information.
classtype	Indicates what type of attack the packet attempted.
payload	
content	Enables Snort to perform a case-sensitive search for specific content (text and/or binary) in the packet payload.
depth	Specifies how far into a packet Snort should search for the specified pattern. Depth modifies the previous content keyword in the rule.
offset	Specifies where to start searching for a pattern within a packet. Offset modifies the previous content keyword in the rule.
nocase	Snort should look for the specific pattern, ignoring case. Nocache modifies the previous content keyword in the rule.
non-payload	
ttl	Check the IP time-to-live value. This option was intended for use in the detection of traceroute attempts.
id	Check the IP ID field for a specific value. Some tools (exploits, scanners and other odd programs) set this field specifically for various purposes, for example, the value 31337 is very popular with some hackers.
dsize	Test the packet payload size. This may be used to check for abnormally sized packets. In many cases, it is useful for detecting buffer overflows.
flags	Test the TCP flags for specified settings.
seq	Look for a specific TCP header sequence number.
icmp-id	Check for a specific ICMP ID value. This is useful because some covert channel programs use static ICMP fields when they communicate. This option was developed to detect the stacheldraht DDoS agent.
post-detection	
logto	Log packets matching the rule to the specified filename.
session	Extract user data from TCP Sessions. There are many cases where seeing what users are typing in telnet, rlogin, ftp, or even web sessions is very useful.

SNORT use

- Three main operational modes
 - Sniffer Mode
 - Packet Logger Mode
 - NIDS Mode
- Operational modes are configured via command line switches
 - Snort automatically tries to go into NIDS mode if no command line switches are given
 - looks for “snort.conf” configuration file in /etc

SNORT use: sniffer mode

- Works much like tcpdump
- Decodes packets and dumps them to stdout
- BPF filtering interface available to shape displayed network traffic
 - BPF = Berkeley packet filter syntax

SNORT use: packet logger mode

- Multi-mode packet logging options available
 - Flat ASCII, tcpdump, XML, database, etc.
 - In order to save the captured packets to disk
- Log all data and post-process to look for anomalous activity

SNORT use: NIDS mode

- Wide variety of rules available for signature engine
 - about 1K in June 2001
 - grow to 3K at May 2005
 - Now exceed 6K rules
- Multiple detection modes available via rules and plug-ins
 - Rules/signature
 - Statistical anomaly
 - Protocol verification

<http://sguil.sourceforge.net>

57

SNORT GUI: Sguil (http query)

SGUIL-0.9.0 - Connected To 192.168.8.250

File Query Reports Sound: Off ServerName: 192.168.8.250 UserName: bamm UserID: 2 2014-11-07 02:17:58 GMT

RealTime Events Escalated Events ES Query 1 ES Query 2 ES Query 3 ES Query 4

Close "query": { "filtered": { } Submit Export Edit

Sensor	Id	Timestamp	Src IP	SPort	Dst IP	DPort	Host	Method	URI	Status
fin-int	87LxGChGT9xlwSw2HpFOA	2014-11-07 01:39:59	192.168.8.72	64889	72.21.91.29	80	ocsp.digicert.com	GET	/MFYwVKADAgEAME0w5z8jMAKGB5sOAwlaBQAEFO1rd3LewDiDoQ...	200
fin-int	-6cpAW6yR62OU9d37XLMeg	2014-11-07 01:38:00	192.168.8.72	64867	54.192.90.166	80	media.pragprog.com	GET	/favicon.ico	200
fin-int	uLy6yfwHQ3mBGs-xt-7nSQ	2014-11-07 01:37:59	192.168.8.72	64868	54.241.5.26	80	www.pragprog.com	GET	/images/covers/190x228/tsgit.jpg	301
fin-int	-GyCuFNITZInghLHEZK0m4w	2014-11-07 01:37:58	192.168.8.72	64867	54.192.90.166	80	media.pragprog.com	GET	/titles/tsgit/images/h1-underline.gif	200
fin-int	sYIGrbwTKT2oY80Vnpe1A	2014-11-07 01:37:58	192.168.8.72	64867	54.192.90.166	80	media.pragprog.com	GET	/titles/tsgit/css/html-only.css	200
fin-int	p7w3F-ATFK2n_0fNRAp6Q	2014-11-07 01:37:58	192.168.8.72	64866	54.192.90.166	80	media.pragprog.com	GET	/titles/tsgit/css/bookshelf.css	200
fin-int	FKUxjrORQeqbQjgnoqr9tQ	2014-11-07 01:37:58	192.168.8.72	64866	54.192.90.166	80	media.pragprog.com	GET	/titles/tsgit/chap-005-extract.html	200

IP Resolution Agent Status Snort Statistics System Msgs User Msgs

☐ Reverse DNS ☒ Enable External DNS

Src IP: Src Name: Dst IP: Dst Name:

Whois Query: ☐ None ☐ Src IP ☒ Dst IP

Organization: Amazon Technologies Inc. (AT-88-Z)
RegDate: 2011-12-09
Updated: 2012-04-02
Ref: http://whois.arin.net/rest/net/NET-54-240-0-0-1

OrgName: Amazon Technologies Inc.
OrgId: AT-88-Z
Address: 410 Terry Ave N.
City: Seattle
StateProv: WA
PostalCode: 98109
Country: US
RegDate: 2011-12-08
Updated: 2014-10-20
Comment: All abuse reports MUST include:
Comment: * src IP
Comment: * dest IP (your IP)
Comment: * dest port
Comment: * Accurate date/timestamp and timezone of activity
Comment: * Intensity/frequency (short log extracts)

View	Field	Value
<input checked="" type="checkbox"/>	host	fin-int
<input type="checkbox"/>	net_name	Int_Net
<input checked="" type="checkbox"/>	_id	uLy6yfwHQ3mBGs-xt-7nSQ
<input checked="" type="checkbox"/>	@timestamp	2014-11-07 01:37:59
<input checked="" type="checkbox"/>	src_ip	192.168.8.72
<input checked="" type="checkbox"/>	src_port	64868
<input checked="" type="checkbox"/>	dst_ip	54.241.5.26
<input checked="" type="checkbox"/>	dst_port	80
<input checked="" type="checkbox"/>	http_host	www.pragprog.com
<input checked="" type="checkbox"/>	http_method	GET
<input checked="" type="checkbox"/>	uri	/images/covers/190x228/tsgit.jpg
<input checked="" type="checkbox"/>	http_status	301
<input type="checkbox"/>	http_referrer	http://media.pragprog.com/titles/tsgit/chap-005-extract.html
<input type="checkbox"/>	http_user_agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.101 Safari/537.36
<input type="checkbox"/>	http_accept_langu...	en-US,en;q=0.8
<input type="checkbox"/>	vendor	suricata

<http://sguil.sourceforge.net>

Other IDS systems (OSS)



ACARM-ng

advanced IDS/IPS system



Bro

network security monitor and NIDS



Fail2ban

log file monitoring and IPS



OSSEC

integrity checker; host-based IDS



Prelude

security inf. and event mgmt.



Sagan

log analysis; correlation engine



Samhain

integrity checker; host-based IDS



Suricata

IDS/IPS; netw. security monitoring

Διαχείριση περιστατικών

Incident response activities

- These include activities pertaining to the
 - Prevention of incidents or attacks from happening in the first place
 - ▶ achieved by securing and hardening the infrastructure
 - Training and educating staff and users on security issues and response strategies
 - Active monitoring and testing their infrastructure for weaknesses and vulnerabilities
 - Sharing of data where and when appropriate with other teams

Incident management plan

- It is an organized, comprehensive, strategic plan to handle computer security incidents from detection to resolution
 - it is also known as *incident management capability*
 - helps avoiding incident response in a reactive, ad hoc manner
 - Implies the end-to-end management for controlling how security incidents will be
 - monitored / detected / responded to / recovered from
- to ensure the organization will continue to meet its operational mission

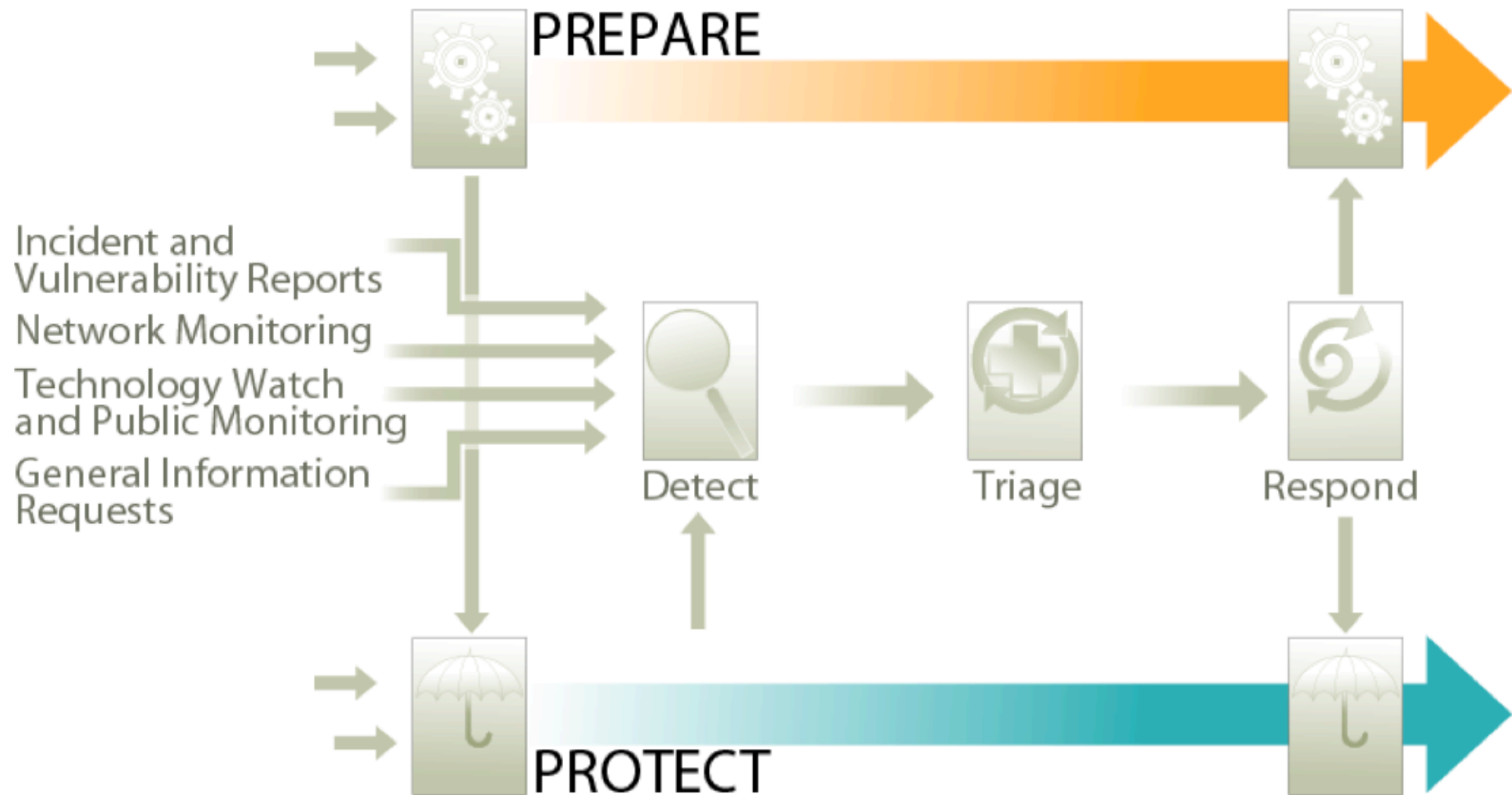
Incident management plan

- The plan must be designed and managed to provide
 - policies and procedures that define and assign the appropriate roles and responsibilities for personnel involved in incident management activities
 - equipment, infrastructure, tools, and supporting materials to protect systems, detect suspicious events and incidents, assist in recovery, and support the resumption of operations
 - qualified staff who are trained to perform consistent, reliable, high-quality incident management functions
 - ▶ i.e. the establishment of an CSIRT

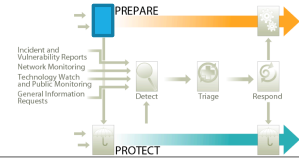
Incident management model

- The model is composed of five high-level processes that are undertaken as part of an incident management plan
 - Prepare (prepare/sustain/improve) – establish and improve a CSIRT
 - Protect (protect infrastructure) – make changes in infrastructure to protect systems or mitigate an ongoing computer security event
 - Detect (detect events) – recognize and report events in real time and look for indicators that might identify future events/incidents
 - Triage (triage events) – categorize, correlate, and prioritize events and assign them to someone for further investigation and response
 - Respond – plan, coordinate, and carry out effective responses

Incident management model

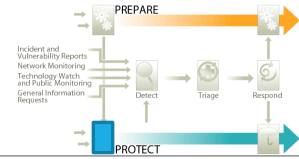


Incident mgmt.: prepare



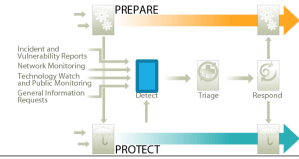
- Plan and implement an initial incident management or CSIRT capability
- Sustain that capability
- Improve an existing capability through lessons learned and evaluation and assessment activities
- Perform a postmortem review of incident management actions when necessary
- Pass off infrastructure process improvements from the postmortem to the **Protect** process

Incident mgmt.: protect



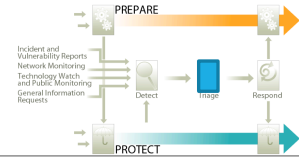
- Implement changes to the infrastructure to stop or mitigate ongoing incidents
 - Or to stop or mitigate the potential exploitation of vulnerabilities
- Implement infrastructure protection improvements due to postmortem reviews or other improvement mechanisms
- Evaluate the infrastructure by performing tasks like proactive scanning, as well as security and risk evaluations
- Pass off to the **Detect** process info on ongoing incidents, discovered vulnerabilities, and security related events

Incident mgmt.: detect



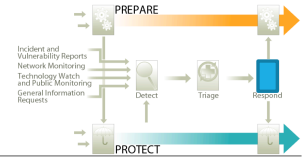
- Notice events and report those events
- Proactively monitor indicators like network monitoring, IDS, or technology watch functions
- Analyze the indicators being monitored
 - To determine any activity that might suggest malicious behavior
- Forward any suspicious event info to the **Triage** process
- Reassign events to areas outside incident mgmt. process
- Close any events not forwarded to the **Triage** process

Incident mgmt.: triage



- Categorize and correlate events
- Prioritize events
- Assign events for handling or response
- Pass on relevant data and info to the **Respond** process
- Reassign events to areas outside incident mgmt. process
- Close any events not forwarded to the **Respond** process
 - or reassigned to other areas

Incident mgmt.: respond



- Analyze the event and plan a response strategy
- Coordinate and provide technical, management, and legal response, which can involve
 - actions to contain, resolve, or mitigate incidents
 - actions to repair and recover affected systems
- Communicate with external parties
- Reassign events to areas outside incident mgmt. process
- Close response and pass lessons learned and incident data to the **Prepare** function for use in a postmortem review

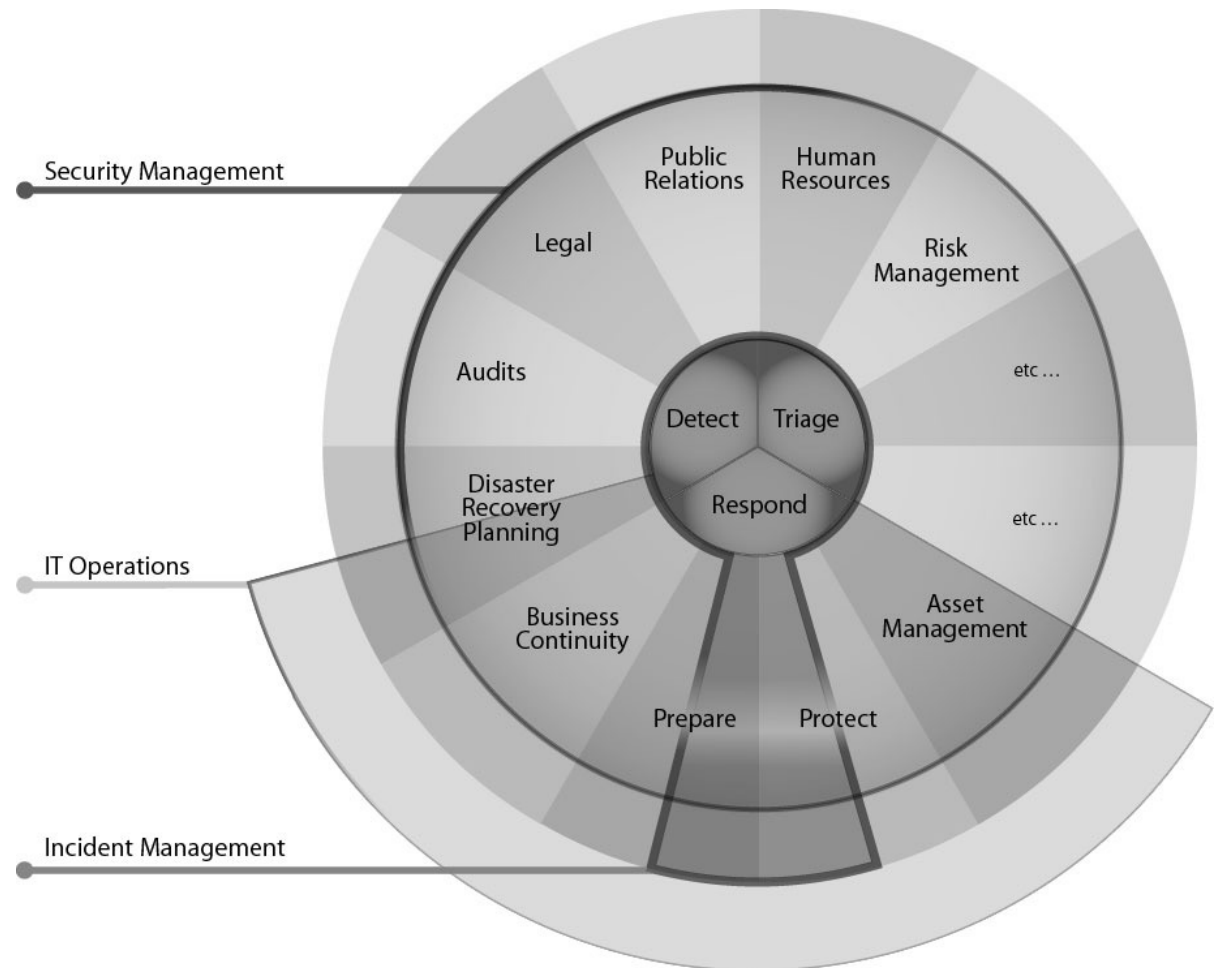
Incident mgmt. workflows

Indicative information
classes

Inform. class	Description
Objectives	Defines what should be accomplished by the successful completion of the process activities
Triggers	Activities initiating the process; can be an event or input
Comp. criteria	Conditions to meet for a successful completion
Policies/rules	Policies, laws, regulations, etc. governing the process
General requirements	Any supporting information that is needed to successfully perform activities associated with this process
Inputs	The required inputs for this process
Input name	The name of the input
Input descr.	A short description of the input, incl. the sending process
Input form	The form of the input (usually verbal, electronic, etc.)
Outputs	The possible outputs of this process
Output name	The name of the output
Output descr.	A short description of the output, incl. its destination
Output form	The form of the output (usually verbal, electronic, etc.)
Subprocess	All of the subprocesses or activities for this process
Subprocess requirements	The requirements for this subprocess, namely what must occur for this subprocess to be successful
Procedures	Procedures to follow by those conducting the subprocess
Key people	The roles of people who may conduct this subprocess
Technology	The types of supporting technology that may be needed to successfully perform this subprocess
Other	Any other relevant items for this subprocess

Incident mgmt. relations

Overlap of security management, incident management, and IT operations



Applying incident mgmt.

- During this process, special attention should be paid for characteristics of the processes, like
 - Missing or poorly defined handoffs
 - Missing or poorly defined aspects of each process activity
 - ▶ e.g. no procedures in place or inadequate staff
 - Bottlenecks in the process
 - Poorly defined activity flows (e.g., too much parallelism, too linear, too many handoffs)
 - Single points of failure

Incident response plan success

- Integrate into the existing processes and organizational structures so that it enables critical business functions
- Strengthen and improve the capability of effectively managing security events
 - keep intact the availability, integrity, and confidentiality of an organization's systems and critical assets, where required
- Support, complement, and link to any existing business continuity or disaster recovery plans

Incident response plan success

- Support, complement, and provide input into existing business and IT policies that impact security
- Implement a command and control structure, clearly defining responsibilities and accountability for actions
- Be part of an overall strategy to protect and secure critical business functions and assets

Προτεινόμενη βιβλιογραφία

- W. Stallings
Cryptography and Network Security: Principles & Practice
7th Ed., Prentice Hall, 2017
- W. Stallings and L. Brown
Computer Security: Principles & Practice
3rd Ed., Prentice Hall, 2015
- M. Bishop
Computer Security: Art and Science
Addison Wesley, 2003