CSCI262 : System Security

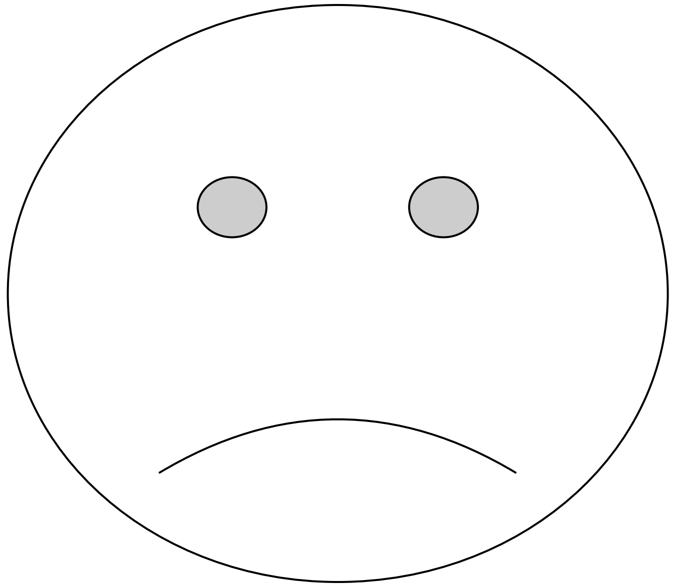
Intrusion Detection System (IDS)

Schedule

- Intrusion detection
- Analysis approaches
- Host-based Intrusion Detection
- Network-based Intrusion Detection
- Snort
- Honeypots

What is intrusion detection?

- We failed … (soft of)
- but at least we might know we failed . . .
- and maybe find what happened . . .
- and maybe stop it happening again . . .



What is intrusion detection?

- Intrusion detection is detecting the *circumvention of a policy*.
  - In particular security policies but we don’t necessarily need to separate them out in particular.
  - This means we failed to enforce a policy → ☹
- Intrusion detection is closely related to system auditing, and underlying both, we need to record what is going on in a system.

- Intrusion detection systems monitor the behaviour within a system with the mindset that there may have been intrusions.
  - That is, the mechanisms for enforcing our policies might not work all the time.
  - Remember access control related to both policies and mechanisms.
- We need to distinguish between intrusions and legitimate behaviour.
- Some types of intrusions are significantly more visible than others.

Categories of attacker

- Attackers roughly fall into 1 of 3 classes.
- **Clandestine**: These try to avoid the intrusion detection or auditing system.
- **Masqueraders**: These pretend to be a legitimate user.
    - So if Oscar has guessed Alice’s password, Oscar can masquerade as Alice.
- **Misfeasors**: These are legitimate users who are misusing the privileges they have.
    - These can often be difficult to determine...

Masquerader vs misfeasor vs legitimate

- All have the password for a legitimate account.
- How can we distinguish between them?
- Maybe on the basis on where or when they log in.
- Likely primarily on the basis of what they do once they are logged in.

A note on IDS vs IPS

- Intrusion detection can be “extended into prevention.”
    - This active extension is then an Intrusion Prevention System (IPS) or Intrusion Detection and Prevention System (IDPS).
- So, for example, an IPS may detect port scanning, which is allowed behaviour, but may be indicative of attack under way, so may actively work to block the attack.

- From Whitman & Mattord, Principles of Information Security, referring to the NIST 800-94 guide:
- *“IPS technologies are differentiated from IDS technologies by one characteristic: IPS technologies can respond to a detected threat by attempting to prevent it from succeeding.”*

“Normal system behaviour”

- Denning (1987) described three characteristics that allow us to determine systems that are behaving normally, or ones that are not.

1) Interactions between subjects and objects follow statistics.
2) Sequences that are likely to circumvent the policies would not be part of typical sequences of behaviour.
3) Processes have a specification set, or a set of allowed actions.

Models

- Behaviour inconsistent with one or several of those characteristics may be taken as evidence of an attempt of an attack.
- A model or strategy allows us to characterise a sequence of states or actions.

- We have two main strategies.
- The first is **anomaly detection**, for when observed behaviour differs from the typical behaviour for a user.
    - This requires statistics on typical user behaviour, for individual users.

THERE'S A RUMOR THAT YOU USE A CHATBOT TO REPLY TO EMAIL WITH USELESS NONSENSE.
@ScottAdamsSays
Dilbert.com
YOU CAN'T PROVE THAT BECAUSE I'VE ALWAYS ANSWERED MY EMAIL WITH USELESS NONSENSE.
7-1-16 © 2016 Scott Adams, Inc. /Dist. by Universal Uclick
THAT WAS DISTURB-INGLY WELL-PLAYED.
IT'S ALL ABOUT CREATING THE BASE CASE.

Dilbert: 01-July-2016

- The second is **signature or misuse based**, when observed behaviour indicates an attempt to inappropriately use resources.
  - This requires that we know typical attack vectors or paths.
- Attack tools such as operating system specific rootkits can, for example, be used to run different versions of standard software, such as ls.
  - These variants would typically breach point 3 of our “normal behaviour”.

Goals of intrusion detection

- Before we look at strategies in more detail it’s useful to identify the goals of intrusion detection systems, and to talk about accuracy.

1) **Detect** all intrusions, internal and external, … meaning it needs to be
2) **Dynamic** in the sense of capable of learning or taking into account current attacks and/or user behaviour. It also needs to be …
3) **Timely**, in the sense of providing information at a point in time at which it is still useful. This isn’t very useful though unless we have …
4) **Clear and Concise Reports** of the results of the analysis. And then there are going to be problems unless the reports are …
5) **Accurate** and known to be accurate. A incorrect report isn’t good, but a correct report that we don’t believe isn’t good either.

False positives and negatives…

- Hopefully you can all remember these terms from authentication, in the context of attempting to distinguish between authorised and unauthorised entities.
  - They are to do with the likelihood of getting a result which is wrong.
  - The interpretation in the context of intrusion detection systems is a little different from that of authentication though.

- A **false positive** is when make a match but "shouldn't have".
- In an intrusion detection system we will be aware of a false positive, which we wouldn't have been with authentication in the context of a false positive.
- A **false negative** is when we don't make a match but "should have".
- This time it's the intrusion detection system where we are unaware, and the authentication system where we are aware.

Anomaly Intrusion detection

- We are going to illustrate this in terms of the user behaviour aspect, although generally it simply the system behaving differently.
  - We could represent these notes in terms of system characteristics…
- We need a way to characterise user behaviour, in a measurable way.
- We can look at measurements associated with individual activities, or with many activities, and we can also look at the actual mix of activities too.

Types of anomaly detection

- **Statistical** :
  - Compares behaviour against measured statistical behaviour.
- **Knowledge based** :
  - Uses an expert system with reference to a set of rules modelling legitimate behaviour.
- **Machine learning** :
  - Uses suitable training data to determine a classification model.

- We are going to focus on statistical anomaly detection.
- Typically the measurables for individual activities can be characterised by time density and event length distributions.
    - How many at a time? And each for how long?

- Measures might include CPU usage, number of processes, number of files opened, typically length of file open time, or process run.

- Depending on the user some of the characteristics may be more significant than others, even within the same system.
- In terms of activity mixes this might be something distinguishing between something like:

edit file, close file, compile file, run process, edit file, close file, compile file, run …   **and**

edit file, edit file, edit file, edit file, edit file, edit file, edit file, …

Statistics: A threshold model

- The simplest statistical model for modelling anomalies is a threshold model.
  - If more than a certain number of something happens an alarm is triggered.

  Or
  - If less than a certain number of something happens an alarm is triggered.
- Login attempts for example.

Mean and standard deviation

- The threshold can be turned into slightly more than a count, particularly if we are dealing with a continuous measure rather than a discrete one.
- Effectively we can specify an anomaly as being a value more than a certain number of standard deviations from the mean.

CATBERT: EVIL DIRECTOR OF HUMAN RESOURCES
EVERY ABSENCE COUNTS AS ONE "OCCURRENCE" WHETHER IT IS ONE DAY OR MORE.
I JUST GOT A HEAD-ACHE. I'LL SEE YOU IN A YEAR. OR AS I LIKE TO CALL IT, ONE OCCURRENCE.
IF SOMETHING IS WORTH HAVING, IT'S WORTH ABUSING.
www.dilbert.com scottadams@aol.com
12-5-06 ©2006 Scott Adams, Inc./Dist. by UFS, Inc.

Dilbert: 05-Dec-2006

Multiple variables: Measures in more detail …

- A collection of measures, (m_1, m_2, …, m_n) and associated weights (w_1, w_2, … w_n).
- Each user has an appropriately determined base profile (M_1, M_2, … M_n).
- To process a system we measure the active profile (μ_1, μ_2, … μ_n).
- We then apply a collection of distance functions D_i to determine $d_i=D_i(M_i,\mu_i)$.

- Our decision can then be based on a threshold, either at the level of individual distances, or in terms of a composite threshold

$$\sum_{i=1}^{n} w_i d_i \leq d_t$$

- It could also be possible to cross-correlate the distance functions.
- The distance functions themselves are functions of the probability distributions associated with the user profile and their actual profile, that is the result of observations.

- Here goes an example …

Event	Average	Stdev	Weight
Logins	4.50	1.25	2
Total time online	287.15	42.12	1
Emails sent	65.40	30.71	1
Orders processed	150.73	20.13	1
Pizza's ordered online	2.03	1.06	0.5

Event	Day 1	Day 2
Logins	7	5
Total time online	300	280
Emails sent	60	75
Orders processed	170	190
Pizza's ordered online	2	4

Calculate $\frac{(7-4.5)}{1.25}$=2

So, logins contribute 2 * 2 = 4.

Aging of data ...

- We shouldn’t really rely heavily on old statistics.
- If we are accumulating data over a significant period of time, and taking it all into account, we should weight the data as a function of time.

- Example –Intrusion Detection Expert System (IDES)
- In IDES subjects: a user, a login session, applications, routers,..., are represented as an ordered sequence of statistics:
  - $(q_{0,j}, \ldots, q_{n,j})$.
  - $q_{i,j}$ is the ith statistic on day j.
  - Metrics are counts or time intervals.
  - The profile for each subject is updated every day.

- Example continued…
  - IDES weights statistics to favour recent behaviour over past behaviour.
  - Let $A_{k,m}$ be the summation of counts making up the metric for the kth statistic on day m.
    - That is the total number of events contributing to the kth statistic up to and including day m.
  - The statistic becomes…

$$q_{k,m+1} = A_{k,m+1} - A_{k,m} + 2^{-r\,t}\, q_{k,m}$$

    - t is the number of log entries or the total time elapsed since time 0.
    - r is a half-life determined through experience.
  - The exponential decay of previous values models the sensitivity to changes in behaviour over time.

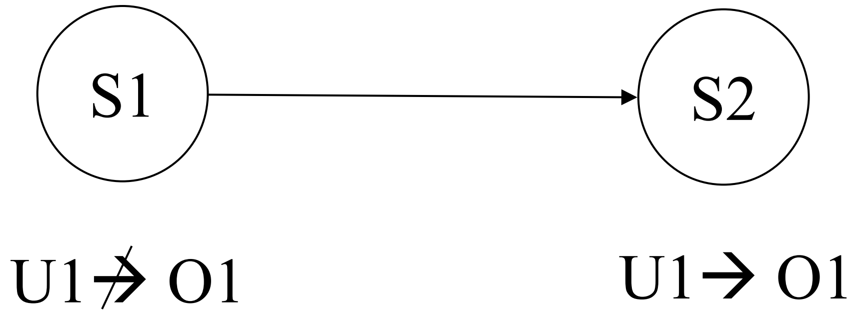
Signature based Intrusion detection

- These look for specifically sequences of events, or resource usages, or some complex condition sets, that describe a known attack.
- Such patterns are sometimes referred to as intrusion signatures.
- Once an attack has been attempted a signature can be extracted and used to detect later instances of the same attack.

- Misuse modelling usually refers to misuse from *within* the system by an authorised user.
- If a set of actions by a user matches to a set of known rules, an intrusion is reported.
- Expert systems are often used to analyse the data.
- Some examples are:
  - Intrusion Detection In Our Time (IDIOT).
    - Monitors Audit logs and looks for sequences of events that correspond to an attack.
  - STAT (State Transition Analysis Tool).
  - Network Flight Recorder (NFR).

STAT (State Transition Analysis Tool)

- This monitors the security state of the system.
- If it goes from a less privileged state to a more privileged state, the way this transition has occurred is monitored.



Network Flight Recorder (NFR)

- This intrusion detect tool takes packets from the network and filters them.
- A backend writes the information generated by these filters to disk.
- Administrators can then query this backend without impacting network performance.

Signature: Misuse versus Specification

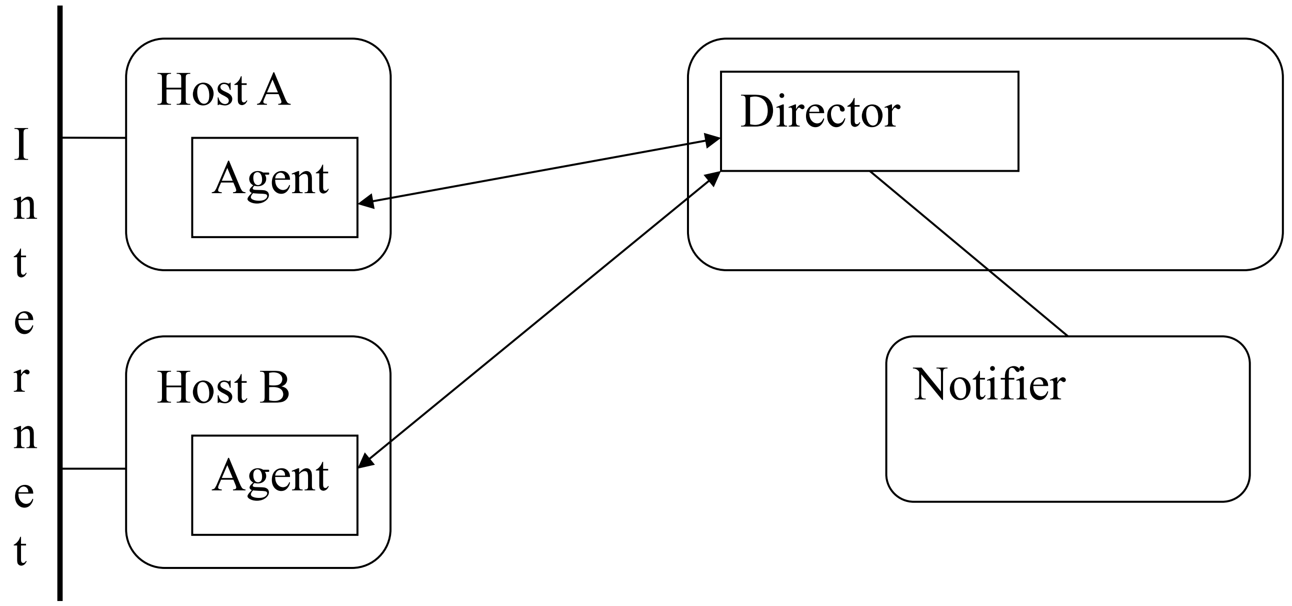
- Misuse modelling looks for states that are known to be bad, specification based-modelling looks for states not known to be good.
  - Black list vs white list.
- Specification-based detection:
  - Determines whether a set of instructions violates the specifications of the system.

- Specification based detection relies on traces or sequences of events.
- A trace policy takes a set of selection expressions and applies them to the system trace of interest.
  - e.g., If a filter is a tuple (proc, prog, host, user) …

    (any, vi, any, john) will produce a list of subjects with program vi and user john.

Architecture of an IDS/IDPS

- The architecture involves:
  - Agents – Gather information from loggers/sensors and likely perform some analysis.
  - A Director – Gather information from agents and perform some analysis.
  - Notifier.



Agents

- An agent collects data from sources, including log files, a network or other processes.
- The agents pre-process information before it gets to the director.
  - For example, agents may transfer failed login attempts from a log, and discard successful logins.
- The director can, however, request more information from the agents.
  - This is particularly important where the correlation between events across agent domains is significant.

How do agents gather information?

- Host-Based information gathering (HIDS/HIDPS):
  - Looks primarily at log files, in the context of a host and likely particular applications.
  - Analyse them to determine what to pass to the director.
    - The events to look for are those relating to the goals of the IDS.
- Network-Based information gathering (NIDS/NIDPS):
  - Monitor network traffic,
  - Detect network-oriented attacks such as DoS.
  - It is impossible if network traffic is encrypted. ☹

Capturing traffic: SPAN or Tap

- Facilitate "passive access to network traffic".
  - They are used in IDS/IPS and sniffers.
- Switched Port Analyzers (SPAN):
  - Switches are used to support traffic flow, typically in the bottom 4 layers.
  - Traffic onto particular ports is copied or mirrored to an analysis dedicated port.

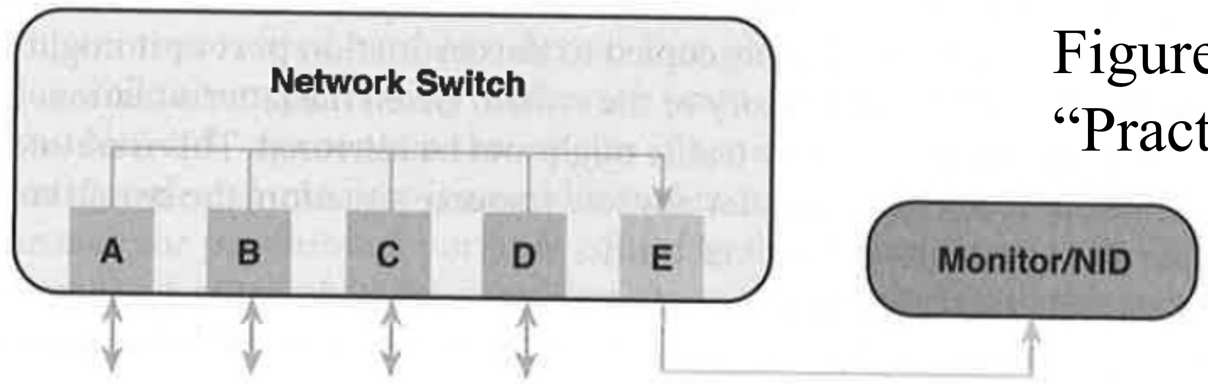


Figure 2.2 From Trost "Practical Intrusion Analysis"

Taps (Terminal Access Points)

- Hardware devices containing multiple ports and include mirroring functionality within them.
  - The interaction of the sniffer/IDS with multiple interfaces means the sniffer/IDS needs to be a more complex entity.

- Combining sources:
  - In Unix, an application level log will be significantly different to a system level log.
  - From the system level view, application logs are insufficient.
  - From the application level view, system logs are insufficient.
  - An agent, or the director, determines the appropriate level, or maps the information to that level.
- One advantage of combining information from multiple sources is the possibility of anomaly detection, used in anomaly based IDS.

- Consider this example of correlating information from multiple logs:
  a. Alice normally logs in during the day to perform system maintenance.
  b. Alice occasionally logs in during the late evening to write reports.
  c. One day, Alice logs in in the late evening and changes the system kernel.

The agent provides logs of both login times, and the commands executed separately.

All appear normal, but looking at them all together we see anomalous behaviour.

Why use Agents?

- Traditional IDS have one point of failure, the director.
- To overcome this we need an IDS where multiple components would function independently, yet still be able to correlate information.
- If one agent has been attacked, the others would still continue to monitor the network.
- What is the cost?
  - There will be an overhead in the communication.

Director

- The director further analyses information, using an analysis engine.
- The director can instruct the agents to:
  - Collect or send more information.
  - Process data differently.
- The director usually requests more information when it detects an attack.
- An agent can obtain information from a set of hosts, in this case it can act as a director with respect to those hosts.
- A director usually runs on a different system, so attackers can’t compromise it at the same time as the system we are trying to protect.

Notifier

- Notifies the appropriate party regarding reports received from the director.
  - Email would be a standard reporting mechanism.
- Graph-oriented user interface.
- The notifier can be responsible for coordinating IDS residing on firewalls to block attacks over the network.
- If an attack is identified the notifier can instruct other IDS's to counteract the attack.

Combining Host and Network Monitoring

- The Distributed Intrusion Detection System (DIDS) was one of the early models, developed in the late 1980's, early 1990's by the US Department of Defence.
  - It is an example of a combined HIDS-NIDS.
- Host only or network only monitoring tends to be ineffective.
  - Logging into a system without a password wouldn't be detected by network monitoring.
  - However, subsequent actions may be detected by host monitoring.

- A DIDS director monitors agents that are attached to hosts as wells as monitoring network traffic.
- Agents scan logs for events of interest and report them to the DIDS director.
- The DIDS director performs analysis using an expert system.
- Note that combining HIDS and NIDS pretty much implies a distributed system.

DIDS Director processing…

The Director has 6 layers in the analysis of the data:

1. Raw data.
2. Relevant entries are extracted.
3. Captures all events related to a single user.
4. Adds contextual information, such as the proximity of other events.
5. Looks at events in relation to context, thus identifies threats and suspicious activities.
6. Assigns a score derived from layer 5 to the security of the network.

Intrusion Response

- Incident Prevention and Handling:
  - If we can we would like to try and prevent the intrusion from succeeding!
  - But we may not be able to, and appropriate handling requires that various mechanisms be in place:
    1. Preparation for an attack.
    2. Identification of an attack.
    3. Containment of the attack.
    4. Eradication of the attack.
    5. Recovery from the attack.
    6. Follow-up to the attack.

- **Preparation for an attack**:
  - Procedures and mechanisms for detecting and responding to attack must be set up before any attacks are detected.
  - Effectively you need to know in advance what to do if an attack is identified.
  - Why do we have fire drills?
- **Identification of an attack**:
  - When an attack is identified the rest of the phases come into play.

Containment of the attack

- We can try and limit access of the attacker to the system:
  - Possibly we could constrain the attacker…
    - Try and prevent the attacker from reaching their goal, if we can identify it.
    - Maybe use a “honeypot” to lead the attacker to another part of the system.
- Passive monitoring:
  - Do nothing, but find out what the attacker is trying to do.
  - Possibly switch to more active measures later.

Eradication of the attack

- Stop the attack:
  - Probably the easiest thing to do is to terminate the network connection or stop the process.
- We could use a firewall to block the attacker.

The last two steps…

- **Recovery from the attack**:
  - We restore the system to a secure state.
- **Follow-up to the attack**:
  - Take some action external to the system against the attacker.
  - Pursue legal action against the attacker.
  - Two methods to identify attacker:
    - Thumb printing.
    - IP Header Marking.
  - Counter-attacking:
    - Only used in exceptional circumstances. ☺

IDS Products

- SNORT: Widely used…
- Anzen Flight Jacket
- Axent/Symantec NetProwler and Intruder Alert
- Check Point IPS-1 (Intrusion Prevention System)
- Cisco Secure IDS
- CyberSafe Centrax IDS
- Endian Firewall
- Enterasys Dragon IDS
- ISS RealSecure
- Network ICE BlackICE Sentry
- Untangle
- See for more updated products: https://www.comparitech.com/net-admin/network-intrusion-detection-tools/

SNORT: Signature based.

- Open source – Freeware, but based at SourceFire where there are quite a few professional developers.
- Provides intrusion detection on Windows, Linux, and Unix.
  - http://www.snort.org/
- The current Windows version: 3
- SNORT allows sets of rules used for detection and logs:
  - Packet information.
  - Alerts.

```
Snort analyzed 523 out of 523 packets, dropping
0(0.000%) packets

Breakdown by protocol:                         Action Stats:
    TCP: 502        (95.985%)                  ALERTS: 12
    UDP: 8          (1.530%)                   LOGGED: 12
   ICMP: 12         (2.294%)                   PASSED: 0
    ARP: 1          (0.191%)
  EAPOL: 0          (0.000%)
   IPv6: 0          (0.000%)
    IPX: 0          (0.000%)
  OTHER: 0          (0.000%)
DISCARD: 0          (0.000%)
```

This is from an old version.

ICMP : Internet Control Message Protocol.

ARP: Address resolution protocol.

EAPOL: EAP over LAN.

```
Wireless Stats:
Breakdown by type:
    Management Packets: 0          (0.000%)
    Control Packets:    0          (0.000%)
    Data Packets:       0          (0.000%)
============================================
Fragmentation Stats:
Fragmented IP Packets: 0          (0.000%)
    Fragment Trackers: 0
   Rebuilt IP Packets: 0
   Frag elements used: 0
Discarded(incomplete): 0
   Discarded(timeout): 0
  Frag2 memory faults: 0
=============================================
TCP Stream Reassembly Stats:
        TCP Packets Used: 502        (95.985%)
         Stream Trackers: 6
          Stream flushes: 2
           Segments used: 14
   Stream4 Memory Faults: 0
```

A SNORT Alert

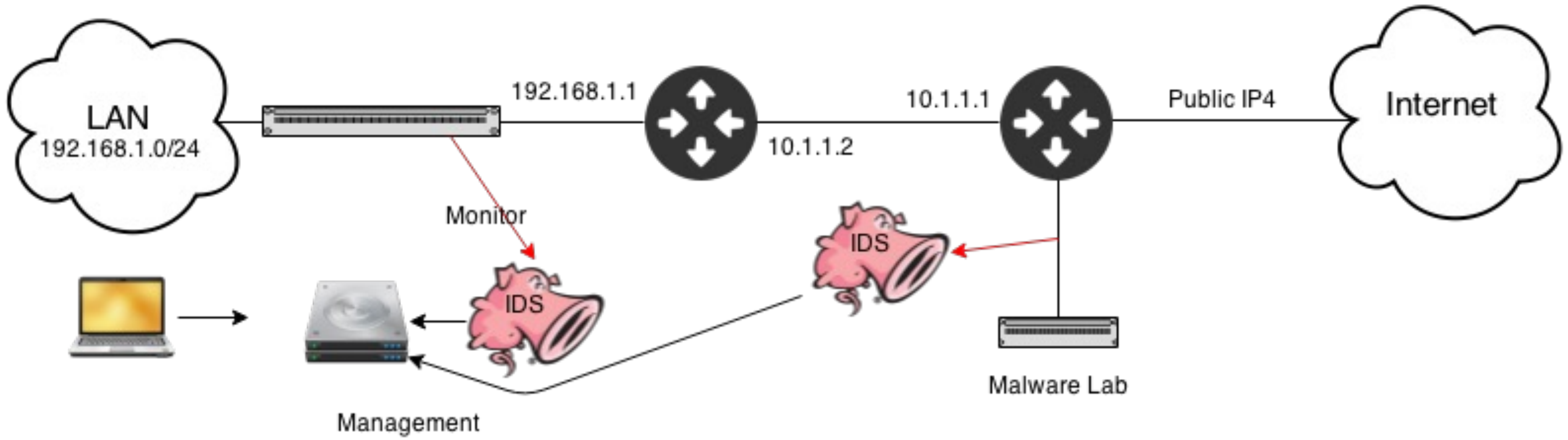
```
[**] [1:620:9] SCAN Proxy Port 8080 attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
08/25-20:28:01.281635 0:50:FC:F1:9E:39 -> 0:60:64:3:DE:7C
type:0x800 len:0x3E
192.168.0.200:1631 -> 130.130.37.205:8080 TCP TTL:128
TOS:0x0 ID:1553 IpLen:20 DgmLen:48 DF
******S* Seq: 0x497BC4F  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
```

Hard to read? → Snorby

- Snorby is a front end for an IDS like Snort.
- https://github.com/Snorby/Snorby
- To set something up at home …

  https://techanarchy.net/home-ids-with-snort-and-snorby/

  - From the website, … their infrastructure …



Honeypots

- Honeypots can divert attackers from a critical system, and/or collect information about the attacker's activity.
- They can encourage the attacker to stay on the system long enough for administrators to respond.
  - We can fill a honeypot with fabricated information designed to appear valuable.
  - A legitimate user of the system will not access that location.
  - We fill the system with sensitive monitors and wait! ☺

Possibly **trap-and-trace**.

- An approximate non-digital analogy corresponds to deliberately leaving the keys in your not-especially valuable car so it gets stolen, …
- … but putting a tracker in too so you can find out where all the stolen cars get taken.
  - Actually putting “invisible” trackers in cars is done (Lojack tags).
- These examples really correspond more closely to related concepts…

… related concepts

- A **honeytoken** is a non-computer honeypot.
  - It could be a fake account, the password for which has been left somewhere by “accident”.
  - … or fake data in a database which wouldn’t normally be accessed.
- A **honeynet** is a collection of two or more honeypots.
  - They can be deployed in such a way as to complement each other, for example they could run under different operating systems but interact with common databases.
- **HoneyMonkey** is the Microsoft Research honeypot.
  - To quote Wikipedia: “ The implementation uses a network of computers to crawl the World Wide Web searching for websites that use browser exploits to install malware on the HoneyMonkey computer.”

Honeypot classification

- **Low interaction honeypot**: Consists of a software package that emulates particular IT services or systems well enough to provide a realistic initial interaction, but does not execute a full version of those services or systems.
- **High interaction honeypot**: Is a real system, with a full operating system, services and applications, which are instrumented and deployed where they can be accessed by attackers.

Honeypot deployment

- Location 1 – outside external firewall
    - Tracking attempts to connect to unsued IP addresses with the network
- Location 2 – DMZ (demilitarised zone)
- Location 3 – fully internal honeypot
    - Can catch internal attacks

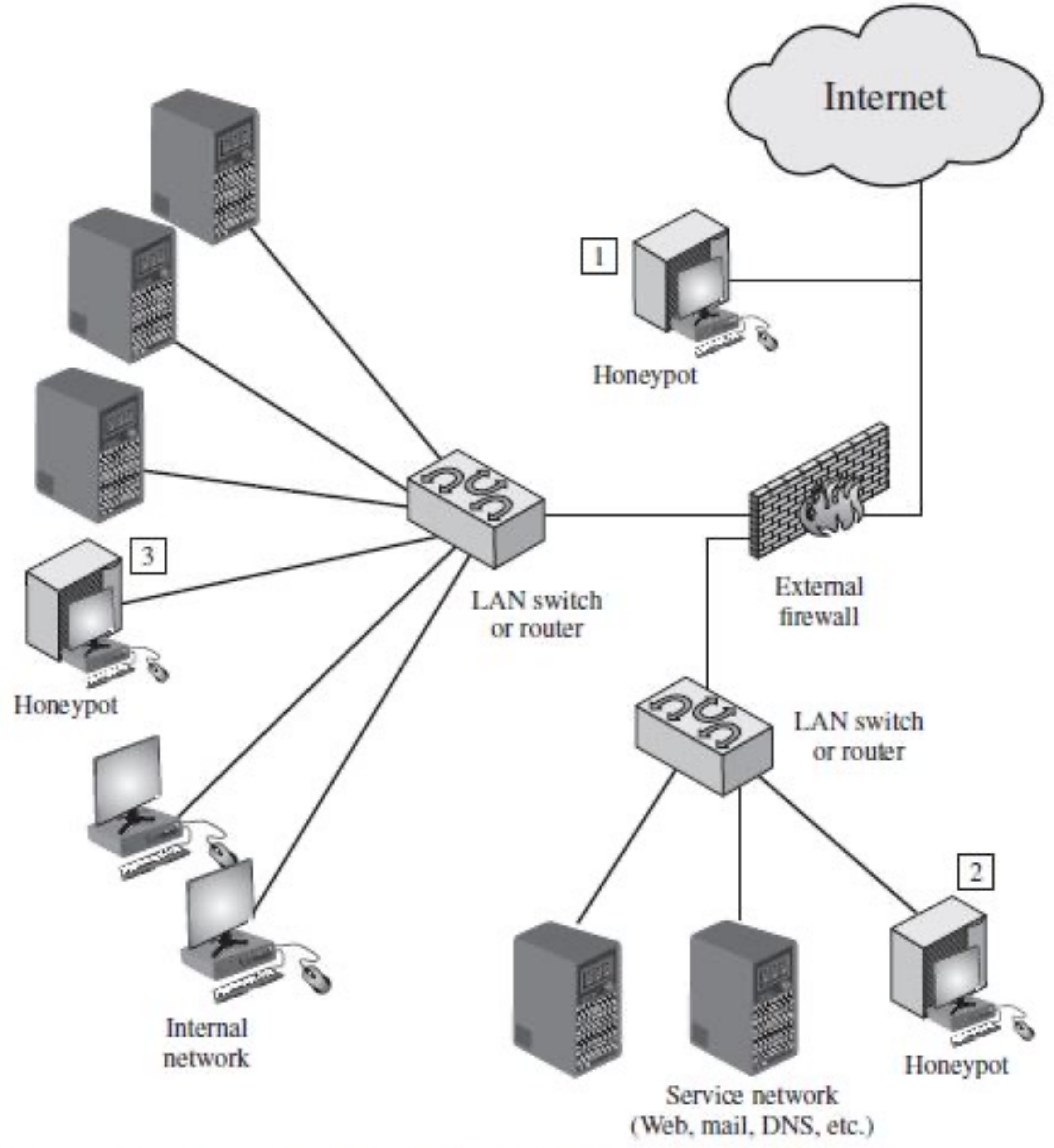


Figure 8.8 **Example of Honeypot Deployment**