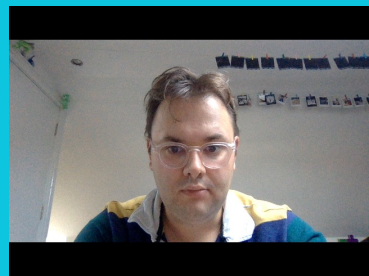
University of BRISTOL

Host-based intrusion detection

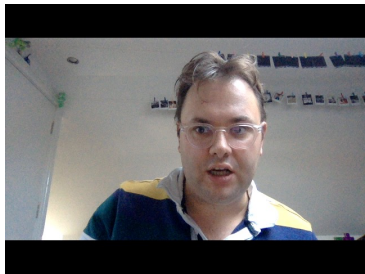
Joseph Hallett



How do we tell when the security mechanisms have failed?

Intrusion detection is a service that monitors a system and looks for unusual or failed attempts to access system resources

- Could be a single event, could be a combination
- Can be probabilistic
- Could be running on the host…
- Could be running on the network…
- Usually attempting to do detection in real-time (or near)



What are we looking for?

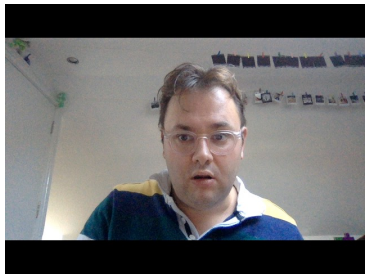
- Failed authentication attempts
- Odd network traffic

- Users running unusual processes
- Users accessing unusual files
  - See also *Tripwire IDS*

- New programs installed
- Known bad programs run

- Changes to configuration
- Anything unusual!

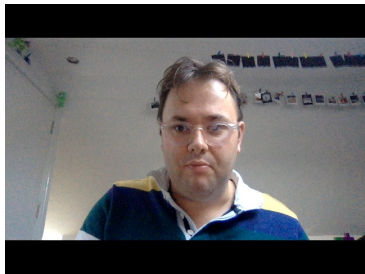
bristol.ac.uk



Types of IDS

Host based
- Runs as a privileged process on the host
- Uses information from the OS/reference monitor

Network based
- Runs either on the host or on the network (often firewall/router)
- Looks at network traffic, who is contacting whom and how often



Types of IDS

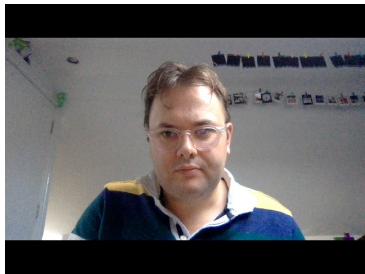
Host Based

Network Based

Signature based
- Identify attacks based on known attack patterns
- E.g. root starts encrypting every file on a system…

Anomaly based
- Identify attacks based on a machine-learning model of what is normal for a given user or process
- If the mail delivery demon suddenly starts accessing files outside of /var/mail...

False positives are problematic (annoy legitimate users)
False negatives also problematic (miss attacks)
- …but can be fed into the rules for *next* time



IDS Goals

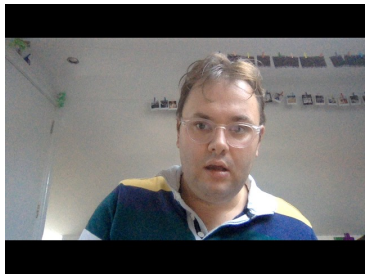
- Run continuously
- Resist attempts to subvert mechanisms

- Don't make the system unusable
  - …in terms of performance overhead
  - …in terms of usability overhead

- Adapt to changes in a system's use
- Allow for reconfiguration

- Scale to work with big systems
- Degrade gracefully
- Fault tolerance

bristol.ac.uk

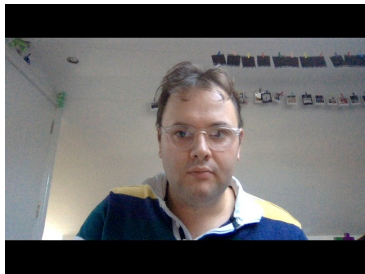


Audit data

What audit data are you going to use to build the IDS?

Native OS data and logs
- System log files, OS provided information (`/proc, /sys`)
- PRO: no extra slow down for the OS as no extra data to collect
- CON: not always enough data or context to be accurate

Add new audit records
- Add extra data-collection mechanisms into the OS
- PRO: collect what you need (including system internal data)
- CON: adds complexity and overhead to the OS



So what do you use?

Anomaly-based can detect new forms of attack (zero days)
• But require the system behaviour be relatively fixed

Signature-based can add rules for new attacks, and change rules for changing system use
• But can only detect what it has rules for

It is a design tradeoff—no *silver bullets* in security!