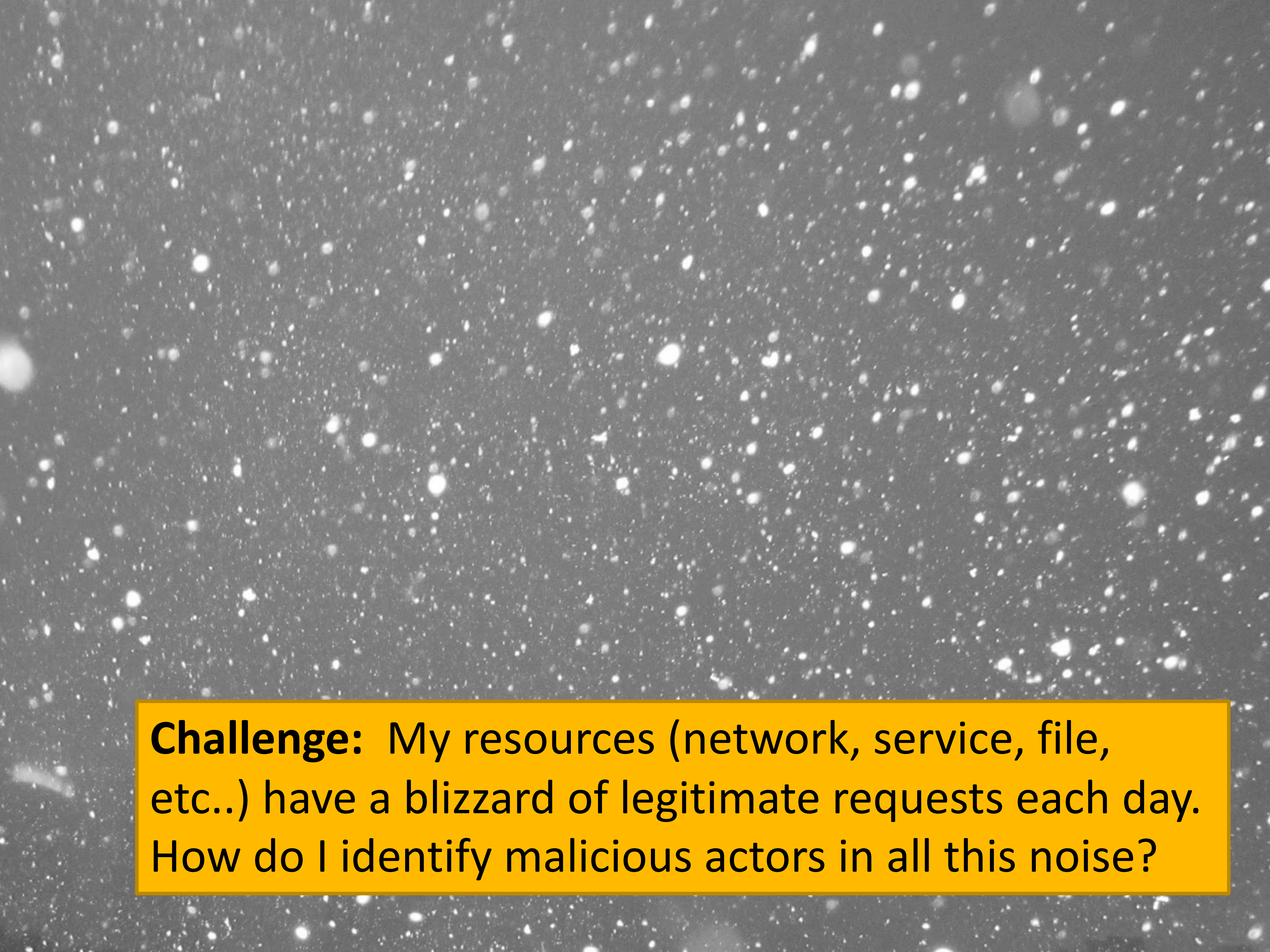
Advanced Computer Networking

CYBR 230 – Jeff Shafer – University of the Pacific

Honeypots



Challenge: My resources (network, service, file, etc..) have a blizzard of legitimate requests each day. How do I identify malicious actors in all this noise?

Honeypots

- A resource that has *no value to legitimate users* but is attractive to attackers
  - Greatly simplifies alerting, as activity on resource is almost always malicious
- **Alert** – Provide early warning of attack (rather than FBI notification 6+ months later)
- **Lure –** Make the attackers waste lots of time here
- **Monitor** – What are the attackers trying to do?
  - Commands entered?
  - Malware uploaded?

Honeypot Use Cases

- **Production systems**
  - *Goal: Protect our current systems*
  - Alert to ongoing attacks that are missed by pattern-based IDS
  - Deterrence (potentially?) if attackers realize they are being monitored
  - Useful for all businesses

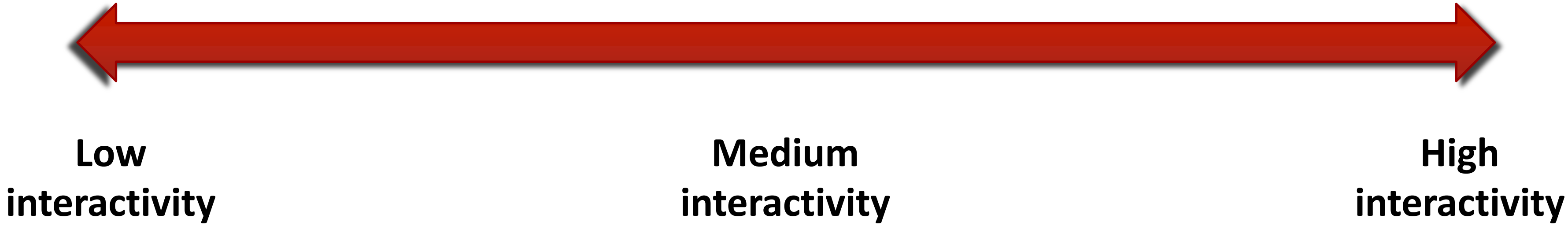
Honeypot Use Cases

- **Research**
  - *Goal: Study attackers*
  - Learn about attacker skill level, tools, motives, origin, …
  - Useful for academics, governments, security researchers, …

Honeypot Interactivity

What kind of interaction can the attacker have with the honeypot?

(in comparison to a *real* vulnerable system)



Low interactivity

Medium interactivity

High interactivity

Low Interaction Honeypot

- Minimal functionality
  - Example: Listen on all TCP ports, accept all connections, and receive data for up to 20 seconds. Send minimal or no replies.
- Pros: Minimal danger to other systems, simple implementation
- Cons: Minimal information learned
  - Source IP, source port, payload sent (if any)

Medium Interaction Honeypot

- System emulates vulnerabilities only
  - Partial simulation of a real system
  - Attacker can’t do much after exploiting vulnerability
- Pros: Reduced danger to other systems
- Cons: Some information learned
  - Attacker is present (source IP)
  - Attacker used specific vulnerability to gain entry
  - What would attacker have done once inside?

High Interaction Honeypot

- Attacker can interact with system at all levels
  - Probe, attack, and compromise
  - Pivot through system for additional attacks
- Equivalent to a real system with hidden monitoring infrastructure
  - Key logging, network logging, file logging, …
  - Data control – Limit where the attacker can go *after* entering honeypot
- Pros: High level of information learned
  - Where are the attackers coming from?
  - What is their skill level?
  - What tools are they using?
- Cons: Risk in letting attacker own our system?
  - Attack the rest of our network?
  - Attack systems outside our organization?
  - Store/distribute illegal content?

Honeypots

- Wide range of possible implementations
  - Dedicated machine
  - Virtual machine
  - Special service on a host
  - Special file on a host
- Never meant for legitimate use
  - Any access is either accidental *or* malicious

Quick and Dirty?

- Q: Why don’t I just install some old unpatched OS and service software in my datacenter? It’ll be attractive to attackers, right?
- A: Method would be possible if the *only* system on your network was the honeypot. But risky in a full datacenter. What if the attackers springboard from the honeypot system to attack legitimate services next?

Thinkst Canary

- Thinkst Applied Research: South African security company
- **Tripwire honeypot**
- Offer a honeypot service (physical hardware, virtual machine, or AWS)
- Paid product ($5k/year for 2 Canaries)

https://canary.tools/

Thinkst Canary

- Configurable to many “personalities”
  - Windows Server 2008, 2003, 10, 8, 7, XP, …
  - Diskstation NAS
  - VMWare ESXi
  - Linux
  - OS X
  - Cisco router, Dell switch
  - Rockwell Automation PLC, Siemens Simatic PLC
  - And more?
- They’re “emulating” these devices to a certain level of fidelity – not really running Windows…

Thinkst Canary

- Configurable with interesting services
  - SSH
  - Telnet
  - SMB (Windows file sharing)
  - Web server (usual suspects, JBoss, VMWare management console, Sharepoint, …)
    - Upload your own fake website, including SSL cert
  - Database
- File shares can be full of fake interesting data
  - Payroll.xls

Thinkst Canary

- Alerts when malicious activity detected
  - SMS, Emails, Slack
  - Visible on external dashboard

Cowrie Honeypot

- **SSH / SFTP honeypot**
  - Fake filesystem (resembles Debian 5.0) with ability to add/remove files
  - Potential (?) for fake file contents, e.g. /etc/passwd
  - SFTP and SCP file uploads/downloads
  - SSH exec commands (`ssh user@host 'cat /etc/passwd’`)
  - SSH tunneling / SSH proxy logging
  - Integration with ELK (ElasticSearch, Logstash, Kibana)

https://github.com/cowrie/cowrie

http://www.micheloosterhof.com/cowrie/

Dionaea Honeypot

Venus flytrap (*Dionaea muscipula)*

- **Malware trap honeypot**
- Identify attackers trying to exploit network server vulnerabilities and capture a copy of the malware they are attempting to run
- Emulates variety of network protocols that attackers are interested in (including vulnerabilities!)
  - FTP, HTTP, Memcache, MySQL, MSSQL, pptp, sib, SMB, …

https://github.com/DinoTools/dionaea

https://jblog.javelin-networks.com/blog/the-honeypot-buster/

https://github.com/JavelinNetworks/HoneypotBuster

Honeypot Buster

- Attempts to detect honeypots (specifically, “honey tokens”)
- Examples
  - Fake domain admin accounts / credentials
    - Set off a red alarm if they are ever used
  - Fake network mapped drives
    - Set off a red alarm if an automated script attempts to access data these drives
  - ….
- There are often signatures that a clever attacker (or script) could use to identify honey tokens as not legitimate

Detectability

- Attackers can obtain the same honeypot software as defenders, and write / distribute fingerprinting scripts to avoid them
- *Constant cat and mouse game*

Other Resources

- "Awesome Honeypots"
  - https://github.com/paralax/awesome-honeypots
  - Curated list
  - *More honeypots (and associated tools) than you ever knew about!*
- HoneyDrive
  - https://bruteforcelab.com/honeydrive
  - Linux distribution with 10 honeypots pre-installed, plus malware, forensics, and network monitoring tools
  - Last updated July 2014 ☹

Project 2

Project 2

- Part 1 - Install and run an existing honeypot
- Can experiment with HoneyDrive but I want final result to be installed from scratch
- Testing location must have unfiltered Internet
  - The lab? AWS?
- Document "interesting findings" as reported by the tool, and explain what you discovered in plain English

Project 2

- Part 2 - Write your own honeypot
- What are you modeling?
  - Software system? IOT appliance?
- Level of interaction? Low, medium, high?
- Level of emulation fidelity?
- How are you going to compare your honeypot to the real thing?
- What is the attacker going to do to or with your honeypot?
- What will you learn from the honeypot about attackers?