Design and methodology of a longitudinal honeypot study

Shreyas Srinivasa, Jens M. Pedersen, *Emmanouil Vasilomanolakis
Aalborg University, Denmark

*Technical University of Denmark
$>:whoami()

- Shreyas Srinivasa
- Ph.D. Fellow – Cybersecurity Research Group, Aalborg University, Copenhagen, Denmark
- Research Interests – Threat Intelligence, Cyber Deception, Internet Security Measurements
- Visiting Scholar at University of Cambridge (Cambridge Cybercrime Centre)
- Prior to Ph.D. – worked in SOC team of a bank in Germany
- Masters from TU Darmstadt, Germany
Regarding the dataset/artifact 😞

- Interaction matters: a comprehensive analysis and a dataset of hybrid IoT/OT honeypots (ACSAC 2022)
- No artifact 😞, thanks to GDPR and legal entanglement around it
- Dataset available as embargo, on request (https://doi.org/10.11583/DTU.21088651)
- Ongoing effort to clear the legal hurdles, Pseudo-anonymization?
- ~5 TB (comp.)
Background

Problem

Design

Methodology

Analysis

Limitations
Honeypots

- deception-based entities that simulate services, gather attack information
- decoys, with a “Know your enemy” concept
- used in defensive security as a trap mechanism
- act as sensors that can be used for malware collection
- study attacker behavior
- insider attacks

classified based on interaction-levels offered to attackers
  - Low – limited simulation of a protocol (application level)
  - Medium – extended simulation, may include a service/device/profile
  - High – actual systems with services configured to work as a honeypot
Value

Any interaction with a “honeypot” system is suspicious

As they are non-production systems, there is no real reason for any interaction with them
## Traditional honeypots

<table>
<thead>
<tr>
<th>Honeypots</th>
<th>Ports &amp; Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kippo</td>
<td>Ports: 22/2222</td>
</tr>
<tr>
<td></td>
<td>Services: SSH</td>
</tr>
<tr>
<td>Cowrie</td>
<td>Ports: 22/2222 23/2323</td>
</tr>
<tr>
<td></td>
<td>Services: SSH, Telnet</td>
</tr>
<tr>
<td>Glastopf</td>
<td>Ports: 80, 8080</td>
</tr>
<tr>
<td></td>
<td>Services: HTTP</td>
</tr>
<tr>
<td>Dionaea</td>
<td>Ports: 80, 443, 21</td>
</tr>
<tr>
<td></td>
<td>Services: HTTP, FTP</td>
</tr>
<tr>
<td>Nepenthes</td>
<td>Ports: 21</td>
</tr>
<tr>
<td></td>
<td>Services: FTP</td>
</tr>
<tr>
<td>Amun</td>
<td>Ports: 23, 21, 80, 36, 143</td>
</tr>
<tr>
<td></td>
<td>Services: Telnet, FTP, HTTP, SMTP, IMAP</td>
</tr>
<tr>
<td>Conpot</td>
<td>Ports: 80, 502, 102</td>
</tr>
<tr>
<td></td>
<td>Services: HTTP, Modbus, S7</td>
</tr>
<tr>
<td>Gaspot</td>
<td>Ports: 100001</td>
</tr>
<tr>
<td></td>
<td>Services: ATG</td>
</tr>
<tr>
<td>MTPot</td>
<td>Ports: 23</td>
</tr>
<tr>
<td></td>
<td>Services: Telnet</td>
</tr>
</tbody>
</table>
Honeynets / Honeyfarms

- Instead of deploying large number of honeypots or honeypots on every network, you simply deploy your honeypots in a single, consolidated location
- Attackers are redirected to the farm, regardless of what network they are on / probing
- Act as sensors and offer telemetry/feed of events
- Source of Threat Intelligence data
- Can be a one consolidated honeypot host or multiple honeypots deployed in diverse locations
● Turning Internet scanning noise into intelligence
● Removing false positives from Internet scanners like Shodan, Censys ...
● Trending vulnerabilities
RQ

Do any operational parameters influence the type of attacks received on a honeypot?

What is the influence of known operational parameters

- Interaction-levels
- Simulation environments
- Deployment infrastructure
- Geo-location
Limitations of current Datasets

- Honeypot datasets are not public (curated)
- Anonymized
- GDPR
- Most honeypots deployed by companies are either in low or medium interaction
- Security corporations have some limitations in what they share, less freedom, low flexibility
# Related work – Honeypot Studies

<table>
<thead>
<tr>
<th>Study</th>
<th>Interaction level</th>
<th>Study period</th>
<th>Geographically distributed</th>
<th>Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honeycloud [7] (2019)</td>
<td>Medium</td>
<td>12 months</td>
<td>Yes</td>
<td>hardware, cloud</td>
</tr>
<tr>
<td>IoTPO [27] (2015)</td>
<td>Low</td>
<td>39 days</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Open for hire [40] (2021)</td>
<td>Low, Medium</td>
<td>1 month</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Muti-faceted Honeypot [52] (2020)</td>
<td>Low</td>
<td>2 years</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Honware [48] (2019)</td>
<td>High</td>
<td>14 days</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Siphon [13] (2017)</td>
<td>High</td>
<td>2 months</td>
<td>Yes</td>
<td>physical, cloud</td>
</tr>
<tr>
<td>Hornet 40 [44] (2021)</td>
<td>Passive</td>
<td>40 days</td>
<td>Yes</td>
<td>cloud</td>
</tr>
<tr>
<td>Picky Attackers [3] (2017)</td>
<td>Medium</td>
<td>4 months</td>
<td>Yes</td>
<td>physical, cloud</td>
</tr>
</tbody>
</table>
Designing a longitudinal honeypot study
-Challenges

- None of the studies had an empirical focus towards all the parameters in the study
- Traditional honeypots are limited in interaction levels (i.e., offer binary interaction, either low or medium or high)
- Some honeypots known to be vulnerable to fingerprinting attacks (*Vetterl et al.)
- Structured attack data collection
- Staleness

To study the influence

- What is the influence of known operational parameters
  - Interaction-levels
  - Simulation environments
  - Deployment infrastructure
  - Geo-location

- Must have multiple interaction levels
- Must simulate multiple protocols (application level)
- Deployed on physical (lab env.) and cloud
- Operational in multiple geo-locations
Background

Problem

Design

Methodology

Analysis

Limitations
RiotPot

- A hybrid-interaction honeypot
- Modular
- Containerized
- Extensibility
- Active noise filter
- Flexible event storage and logging

https://github.com/aau-network-security/riotpot
# Related work – Honeypot Studies

<table>
<thead>
<tr>
<th>Study</th>
<th>Interaction level</th>
<th>Study period</th>
<th>Geographically distributed</th>
<th>Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honeycloud [7] (2019)</td>
<td>Medium</td>
<td>12 months</td>
<td>Yes</td>
<td>hardware, cloud</td>
</tr>
<tr>
<td>IoTpot <a href="2015">27</a></td>
<td>Low</td>
<td>39 days</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Open for hire [40] (2021)</td>
<td>Low, Medium</td>
<td>1 month</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Muti-faceted Honeypot <a href="2020">52</a></td>
<td>Low</td>
<td>2 years</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Honware [48] (2019)</td>
<td>High</td>
<td>14 days</td>
<td>No</td>
<td>physical</td>
</tr>
<tr>
<td>Siphon <a href="2017">13</a></td>
<td>High</td>
<td>2 months</td>
<td>Yes</td>
<td>physical, cloud</td>
</tr>
<tr>
<td>Hornet 40 <a href="2021">44</a></td>
<td>Passive</td>
<td>40 days</td>
<td>Yes</td>
<td>cloud</td>
</tr>
<tr>
<td>Picky Attackers [3] (2017)</td>
<td>Medium</td>
<td>4 months</td>
<td>Yes</td>
<td>physical, cloud</td>
</tr>
<tr>
<td>RioTPot (2022)</td>
<td>Low, High, Hybrid</td>
<td>3 months</td>
<td>Yes</td>
<td>physical, cloud</td>
</tr>
</tbody>
</table>
Design - Longitudinal Study

- 3 Interaction levels - Low, High, Hybrid
- 2 Deployment environments - lab, cloud
- 12 independent honeypot hosts per interaction level
- 4 geographical locations - Denmark(Lab), Germany, New York City, Singapore
- 6 application protocols – Telnet, SSH, HTTP, MQTT, Modbus, CoAP
- Comparison with 1 medium interaction honeypot – Conpot
- 3 months of evaluation
# Design - Longitudinal Study

<table>
<thead>
<tr>
<th>Host</th>
<th>Environment</th>
<th>Geo-Location</th>
<th>Interaction-level</th>
<th>Protocols Emulated</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>Lab</td>
<td>Denmark</td>
<td>High</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R2</td>
<td>Lab</td>
<td>Denmark</td>
<td>Low</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R3</td>
<td>Lab</td>
<td>Denmark</td>
<td>Hybrid</td>
<td>High - SSH, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Low - Telnet, HTTP</td>
</tr>
<tr>
<td>C1</td>
<td>Lab</td>
<td>Denmark</td>
<td>Medium</td>
<td>Telnet, SSH, HTTP, Modbus, S7</td>
</tr>
<tr>
<td>R4</td>
<td>Cloud</td>
<td>New York City</td>
<td>High</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R5</td>
<td>Cloud</td>
<td>New York City</td>
<td>Low</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R6</td>
<td>Cloud</td>
<td>New York City</td>
<td>Hybrid</td>
<td>High - SSH, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Low - Telnet, HTTP</td>
</tr>
<tr>
<td>C2</td>
<td>Cloud</td>
<td>New York City</td>
<td>Medium</td>
<td>Telnet, SSH, HTTP, Modbus, S7</td>
</tr>
<tr>
<td>R7</td>
<td>Cloud</td>
<td>Frankfurt</td>
<td>High</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R8</td>
<td>Cloud</td>
<td>Frankfurt</td>
<td>Low</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R9</td>
<td>Cloud</td>
<td>Frankfurt</td>
<td>Hybrid</td>
<td>High - SSH, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Low - Telnet, HTTP</td>
</tr>
<tr>
<td>C3</td>
<td>Cloud</td>
<td>Frankfurt</td>
<td>Medium</td>
<td>Telnet, SSH, HTTP, Modbus, S7</td>
</tr>
<tr>
<td>R10</td>
<td>Cloud</td>
<td>Singapore</td>
<td>High</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R11</td>
<td>Cloud</td>
<td>Singapore</td>
<td>Low</td>
<td>Telnet, SSH, HTTP, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td>R12</td>
<td>Cloud</td>
<td>Singapore</td>
<td>Hybrid</td>
<td>High - SSH, MQTT, Modbus, CoAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Low - Telnet, HTTP</td>
</tr>
<tr>
<td>C4</td>
<td>Cloud</td>
<td>Singapore</td>
<td>Medium</td>
<td>Telnet, SSH, HTTP, Modbus, S7</td>
</tr>
</tbody>
</table>

Table 2: Experimental setup overview
Background

Problem

Design

Methodology

Analysis

Limitations
RiOTPot – adapting for the study

- Interactive setup and configuration shell
- Enhancing the emulation of SSH, Modbus, HTTP, MQTT, CoAP protocols
- Inclusion of verified docker images for the high-interaction emulation
- pcap analysis with Arkime and a pcap repository for extended packet-level capture and analysis
Lab Setup (Denmark)
Cloud Setup

Cloud Setup

New York City

R4

High Interaction Mode

R5

Low Interaction Mode

R6

Hybrid Interaction Mode

C2

Medium Interaction Mode

Frankfurt

R7

High Interaction Mode

R8

Low Interaction Mode

R9

Hybrid Interaction Mode

C3

Medium Interaction Mode

Singapore

R10

High Interaction Mode

R11

Low Interaction Mode

R12

Hybrid Interaction Mode

C4

Medium Interaction Mode

Cybersecurity Research Group
cyber.aau.dk
Background

Problem

Design

Methodology

Analysis

Limitations
Dataset

- A comprehensive dataset of *pcaps* and events in database

- The database schema contains
  - Source IP address (attacker)
  - Destination IP addresses (honeypots, anonymized)
  - Source IP ports
  - Destination IP ports
  - Timestamps
  - Geolocation of the attacker IPs
  - Interaction level of the honeypots and protocols (where the attack event was observed)
  - Deployment environment information of the honeypots (Cloud/Lab)
  - IP layer traffic and flags
  - Transport layer traffic and flags
  - Application layer data transmitted
Data analysis

- The analysis was done on events recorded in json format in MongoDB
- The packet level inspection was done with Arkime
- The metadata for further analysis was requested from Greynoise
Combing/breakdown

Data Indexing

Aggregation

Visualization

Finding Anomalies

Metadata collection

DPI

Patterns
**Parameter:** Geo-location, city, interaction level, events

- Sphere size denotes the number of daily events per day by interaction-level
- Lowest received: 743, highest: 13,287
- The lab instances received lower malicious events
- The Frankfurt instances (cloud) received the highest traffic overall
Parameter: Geo-location, lowest-highest, interaction-level

1. Highest events recorded in Frankfurt, with High Interaction
2. Lowest events recorded in lab deployment, with Low-interaction
3. Regardless, the High-interaction deployments received the highest events
Background

Problem

Design

Methodology

Analysis

Limitations
Limitations

- One Lab deployment environment; uneven comparison with the cloud deployments
- Limited to 4 cities in 3 continents
- 6 protocols
- We consider each connection as an event, entailing limitations in terms of over-counting
- Not in Netflow format (flexible integration)
- Sharing limitations; GDPR issues in Europe (IP is considered sensitive information)
Failures

- Hosting “vulnerable” instances is tricky
- The National CERTS don’t want vulnerable instances around
- Also, in the cloud (ingress, egress rules)
- Cost!
- Monitoring
Summary

- Honeypots are still an effective tool; if configured carefully
- The parameters play an important role in honeypots and honeypot studies
- Configuring the parameters based on studies provide a broader overview of the attack landscape

- Supplementary findings
  - High-interaction honeypots receive higher attack events
  - Location-specific attacks observed
  - There is an increase in “scanning-service” traffic, many new services observed
Lessons learnt

- Deploying, managing and operating honeypots is challenging
- Attackers could exploit honeypots to launch attacks
- Deception–based systems are a great resource, however you must have a strategy and look for what you need
- Threat Hunting is a tedious task, especially when you have billion events per day
- The dataset is precious; however, the GDPR issues make the public sharing challenging – Open Question!
References


Acknowledgement

- Dr. Richard Clayton
- Dr. Alice Hutchings
- Cambridge Cybercrime Centre, University of Cambridge
- Rich, curated datasets on Internet scanning, honeypots, DarkWeb, DeepWeb and more.
More from our research group
HosTaGe - an Interactive, mobile-based honeypot
Contact

- Shreyas Srinivasa
- shsr@es.aau.dk
- https://sastry17.github.io
- Datasets on Selective Internet Scanning, Honeypots, Darkweb (marketplaces, forums)