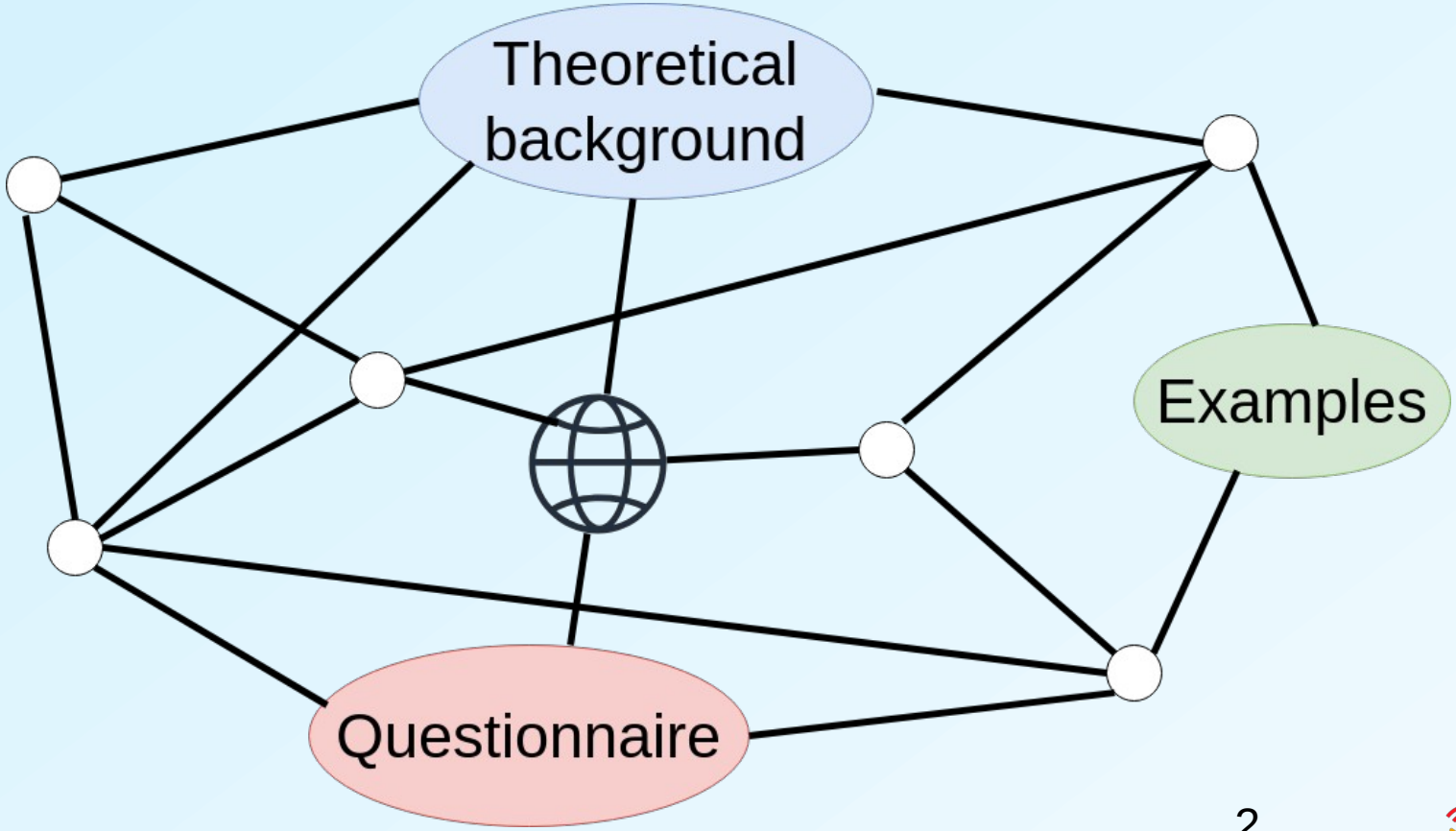
SAND5G

Honeypot Training – Understanding Cybersecurity Traps

Detecting & Analyzing Cyber Threats

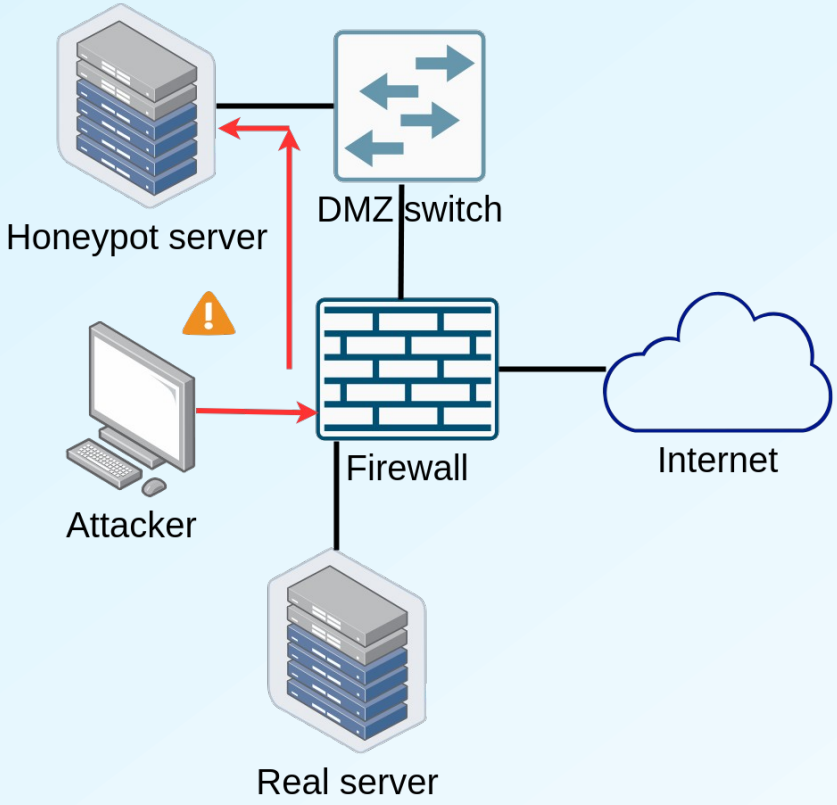
Dionysia Varvarigou

Contents



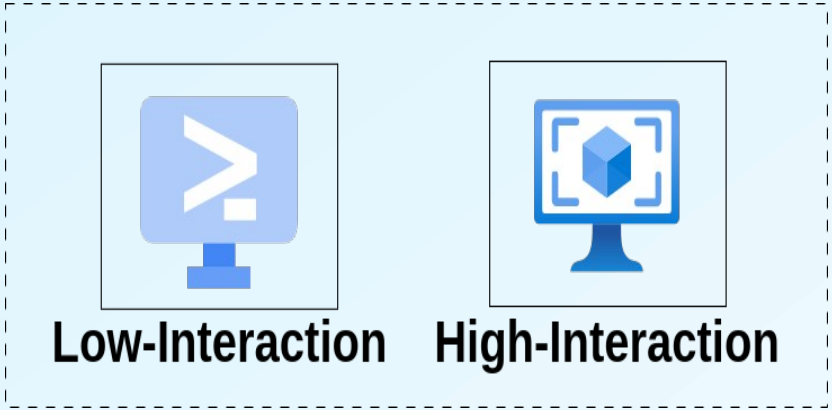
Introduction to Honeypots

- Definition: A honeypot is a decoy system designed to attract and analyze cyber attackers [1]
- Example : A honeypot could mimic an open ssh port, a frequent attack of criminals who want to get access to a system
- Purpose [2]:
  - Detect cyber threats
  - Gather intelligence on attacker behavior
  - Divert attackers from real systems



Types of Honeypots

- Low-Interaction: Simulate some portion of the operating system, such as the network stack, while focusing on services that cannot be utilized by the intruder to adventure the real system [3]
- High-Interaction: complete production similar systems that have a full set of services and permit an intruder a great deal of scope throughout the intrusion [4]
- Production vs. Research:
  - Production: Protects real assets
  - Research: Collects attack intelligence



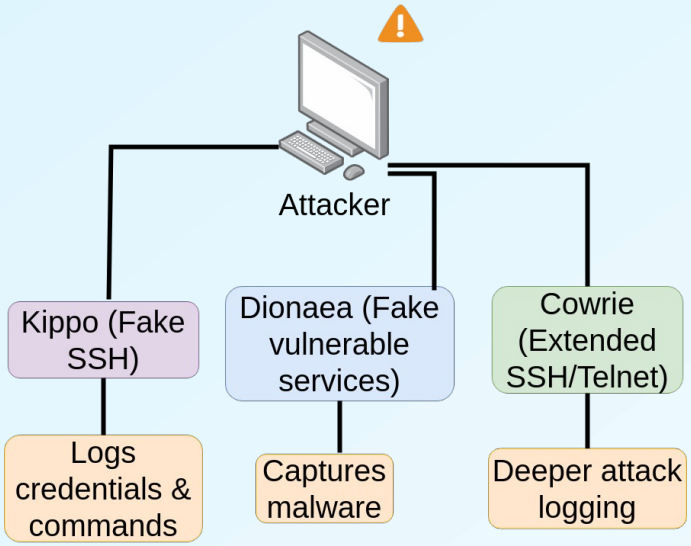
How Honeypots Work

•Honeypot operation follows three key steps [5]:

- Deployment: Placed in a network to attract attackers
- Data Collection: Logs attacker activity
- Analysis & Response: Used to improve security strategies

•Example : A company deploys a fake login page as a honeypot, simulating a real website. The honeypot is accessible on the internet but isolated from critical systems. An attacker discovers the login page and attempts an SQL injection attack. The honeypot logs every request, including the attack pattern and the attacker's IP address. The security team analyzes the logs, identifying that the attacker was trying to exploit SQL vulnerabilities. So they procceed as follows :

- They update firewall rules to block the attacker's IP
- They improve real database security by fixing vulnerabilities
- The honeypot's data is used to enhance intrusion detection rules



Advantages of Honeypots

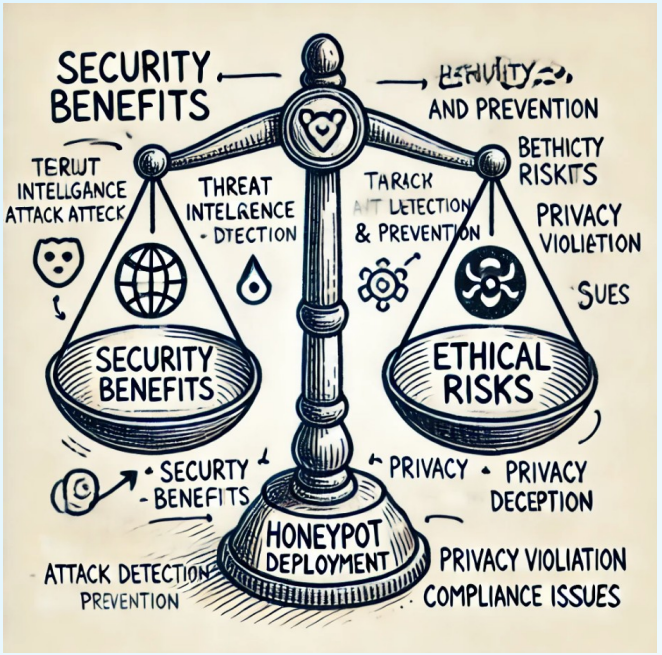
- Honeypot advantages as found in [6] :
  - Data Value : Honeypots gather precise data which is easy to understand in order to facilitate easy analysis of data.
  - Resources : Honeypots do not face the problem of resource exhaustion unlike other security mechanisms, since they capture data directed to them and only.
  - Simplicity : They do not require high end algorithms and configurations. Also they are much easy to use. Simply deploying and monitoring is what we require to do.
  - Return on Investment : Honeypots are quite valuable as it quickly captures the malicious activities. It reflects the security mechanism level of the system.
  - Reduce false positives : Various security mechanisms provide a potential amount of false positive alert messages but honeypots do not provide false positives as it is mostly accessed by the intruders.

Limitations of Honeypots

- Honeypot limitations as found in [6] :
- Coverage Limitation :
  - Issue: Around 95% of locally detected alerts do not have a corresponding reference in external databases
  - Cause: Honeypots, despite their ability to interact with attackers, do not cover a full range of attack scenarios
  - Example: Some attacks, such as Web-Client Windows Media Player directory traversal exploits, require deeper interaction, which low to medium-interaction honeypots cannot always provide
  - Impact: This lack of coverage complicates correlation and enrichment processes, requiring the use of generalization methodologies
- Unfilled Attributes & Missing References :
  - Issue: Many threat datasets contain incomplete attributes, making automated reasoning and data correlation difficult
  - Example: A study on 73,699 threats from HARMUR showed that 85% lacked references to content objects, which provide in-depth details on the threat type and its vulnerabilities
  - Impact: Without temporal attributes, it becomes impossible to apply effective filtering techniques, hindering analysis
- Lack of Standardized Threat Data Representation :
  - Issue: Threats are often categorized using generic labels (e.g., phishing, virus, browser exploit) without deeper classification
  - Problem: This limits qualitative analysis and prevents effective semantic filtering
  - Example: A honeypot may detect a threat on a specific server, but due to ambiguous labeling, it is unclear whether this alert correlates with a larger attack phenomenon
- Cross-Reference Limitations
  - Issue: Honeypot databases often do not share data or coordinate effectively, making deep threat analysis challenging
  - Example: A malware infection reported in HARMUR may have valuable characterization details stored in SGNET, but there is no automated way to correlate this data
  - Impact: The lack of global threat visibility makes it difficult to track infection sources, evolution over time, and exploit characteristics

Legal & Ethical Considerations

- Ethical considerations as found in [7] :
- Privacy Concerns :
  - Honeypots must avoid collecting personally identifiable information (PII). Traditional honeypots do not store user data, but social honeypots may inadvertently capture private communications.
  - Example: Social networks contain private posts that users do not expect to be monitored.
- Use of Deception :
  - Honeypots rely on deception to attract attackers, raising ethical questions. In social honeypots, users may interact with fake accounts without realizing it.
  - Example: The "emotional contagion" Facebook study caused controversy when users were unknowingly manipulated.
- Legal & Compliance Risks :
  - Honeypots must align with data protection laws and ethical research standards. GDPR and similar regulations restrict data collection without consent.
  - Institutional Review Boards (IRBs) assess human subjects research risks in ethical cybersecurity studies.
- Responsible Research Practices :
  - Transparency, accountability, and minimizing harm are key principles. Researchers should document methodologies and ensure compliance with ethical standards (e.g., Menlo Report).
  - Where possible, honeypots should be designed to reduce unintended harm and avoid interfering with law enforcement operations.

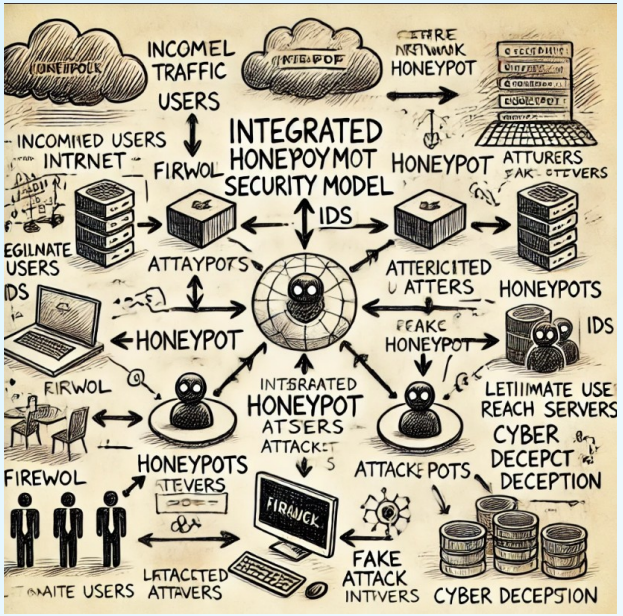


Deployment Strategies

- Defending Against DDoS Attacks with Honeypots [8]: DDoS attacks overwhelm networks, especially critical DMZ servers. Proven effective through simulations on Internet-type topologies. Honeypot-based multi-layer defense:
  - Detection: Identifies and tags attack traffic in real time
  - Dynamic Routing: Deploys extra server nodes for real users while redirecting attackers to honeypots
  - Containment: Keeps attackers engaged in isolated environments, ensuring network stability

- Honeypots in Intrusion Detection & Prevention (IDPS) [9] : Combines honeypots with IDS to analyze threats in real-time. Hybrid honeypot approach:
  - Uses virtualization to simplify deployment
  - Detects zero-day attacks and reduces false positives compared to traditional IDSs
  - Provides real-time traffic visualization for better security monitoring

- Advanced Cyber Deception with Honeypots : Misleads attackers to disrupt their reconnaissance and waste their resources. Enhances security by confusing, delaying, and countering cyber threats. Three methodologies:
  - Dynamic Honeypots adapt to evolving network environments [10]
  - Game-Theoretic Deception optimizes honeypot placement using AI-based decision-making [11]
  - Active Directory Honeypots defend high-value targets using strategic attacker-blocking techniques [12]



Honeypots vs. Other Security Tools

Feature	Honeypot	IDS/IPS	Firewall
Detects Attacks	✅	✅	❌
Engages Attacker	✅	❌	❌
Blocks Attacks	❌	✅	✅
Provides Threat Intelligence	✅	✅	❌

Future of Honeypots

- Future insights according to [13] :
  - Advanced Attack Detection: Enhanced Monitoring deals with developing sophisticated monitoring tools within honeypots to identify and analyze complex attack patterns in real-time.
  - Extensive Data Analysis: Threat Pattern Analysis utilizes big data analytics to process information gathered by honeypots, aiming to uncover emerging cyber threat trends and patterns.
  - Automation of Data Analysis: AI Integration implements artificial intelligence and machine learning algorithms to automate the analysis of data collected from honeypots, improving response times and accuracy.
  - Honeypot Security in Emerging Technologies: Adaptation to New Environments are capable to investigate the deployment and security of honeypots within novel technological landscapes, such as Internet of Things (IoT) ecosystems and cloud infrastructures.
  - Continuous Risk Evaluation: Dynamic Assessment regularly assess and update honeypot configurations to ensure their effectiveness against evolving cyber threats.
- Additional future insights according to [14] :
  - Deception Techniques: Advanced Mimicking is able to design honeypots that closely replicate legitimate systems to effectively attract and engage attackers. Honeytoken Baiting deploys deceptive data tokens to lure attackers and monitor their activities.
  - Honeynet Optimization: Dynamic Configuration adjusts honeynet setups in real-time to respond to detected threats and changing network conditions.

References

1.Spitzner, Lance. "The honeynet project: Trapping the hackers." IEEE Security & Privacy 1.2 (2003): 15-23

2.Bringer, Matthew L., Christopher A. Chelmecki, and Hiroshi Fujinoki. "A survey: Recent advances and future trends in honeypot research." International Journal of Computer Network and Information Security 4.10 (2012): 63

3.Shiue, L., Kao, S.: Countermeasure for detection of honeypot deployment. In: IEEE International Conference on Computer and Communication Engineering, ICCCE 2008, pp. 595–599. IEEE (2008)

4.Mohammadi, S., Nikkhahan, B.: A fault tolerance honeypots network for securing E-government. In: IEEE International e-Conference on Advanced Science and Technology, AST 2009, pp. 13–17. IEEE (2009)

5.Provos, Niels. "A Virtual Honeypot Framework." USENIX Security Symposium. Vol. 173. No. 2004. 2004

6.Titarmare, Neha, Nayankumar Hargule, and Anand Gupta. "An overview of honeypot systems." International Journal of Computer Sciences and Engineering 7.2 (2019): 394-397

7.Dittrich, David. "The ethics of social honeypots." Research Ethics 11.4 (2015): 192-210

8.A. Sardana and R. C. Joshi, "Autonomous dynamic honeypot routing mechanism for mitigating DDoS attacks in DMZ," 2008 16th IEEE International Conference on Networks, New Delhi, India, 2008, pp. 1-7, doi: 10.1109/ICON.2008.4772623

9.Muhammet Baykara, Resul Das, A novel honeypot based security approach for real-time intrusion detection and prevention systems, Journal of Information Security and Applications, Volume 41, 2018, Pages 103-116, ISSN 2214-2126

10.I. Kuwatly, M. Sraj, Z. Al Masri and H. Artail, "A dynamic honeypot design for intrusion detection," The IEEE/ACS International Conference onPervasive Services, 2004. ICPS 2004. Proceedings., Beirut, Lebanon, 2004, pp. 95-104, doi: 10.1109/PERSER.2004.1356776

11.A. H. Anwar, C. A. Kamhoua, N. O. Leslie and C. Kiekintveld, "Honeypot Allocation for Cyber Deception Under Uncertainty," in IEEE Transactions on Network and Service Management, vol. 19, no. 3, pp. 3438-3452, Sept. 2022

12.H. Q. Ngo, M. Guo and H. Nguyen, "Catch Me if You Can: Effective Honeypot Placement in Dynamic AD Attack Graphs," IEEE INFOCOM 2024 - IEEE Conference on Computer Communications, Vancouver, BC, Canada, 2024, pp. 451-460

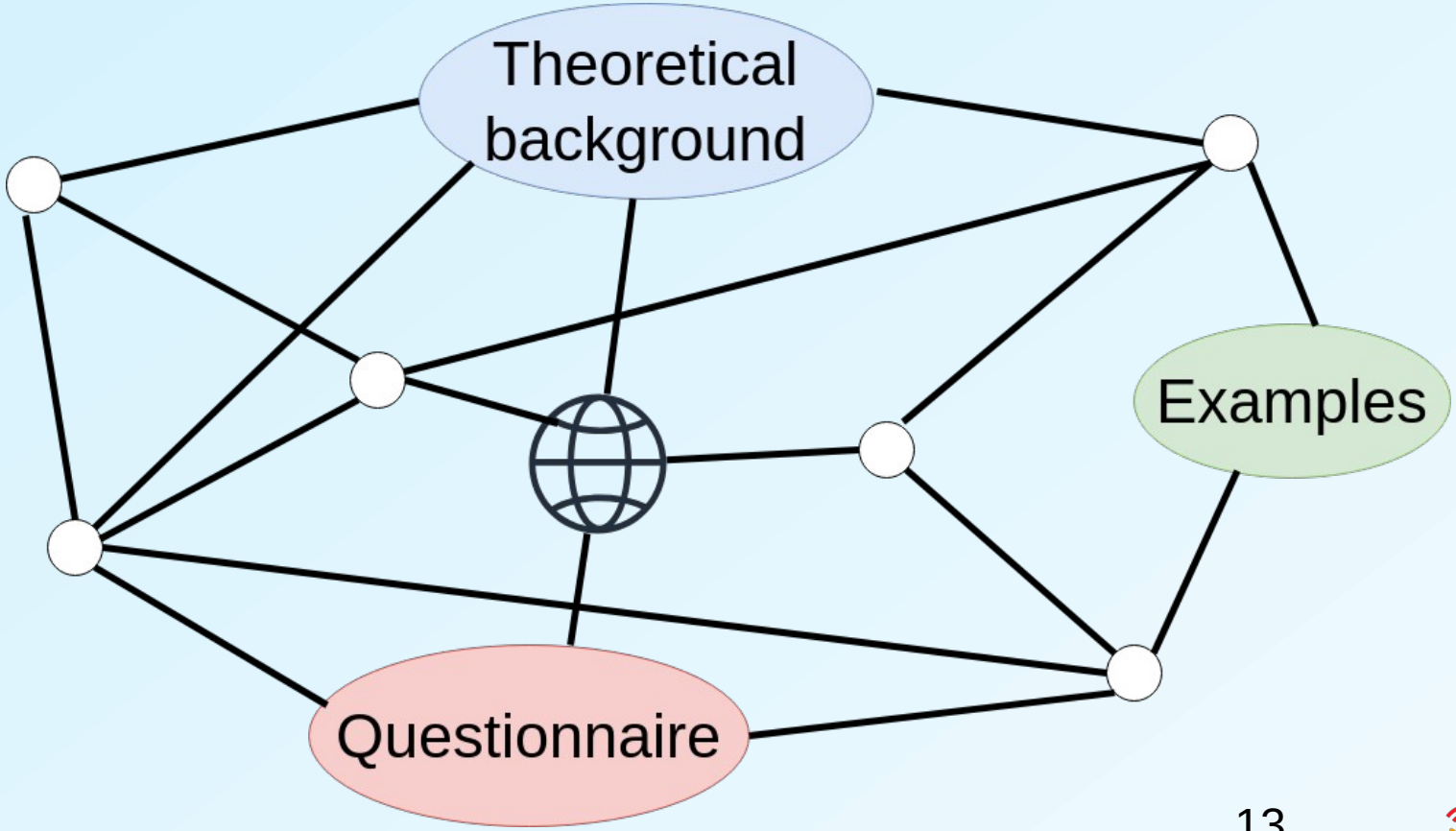
13.DOSKU, Lediano. "Honeypots, for a more secure network." Editorial Board (2024): 66

14.Javadpour, Amir, et al. "A comprehensive survey on cyber deception techniques to improve honeypot performance." Computers & Security (2024): 103792

15.Stock/vector icons. Icons used in figures sourced from The Noun Project, Freepik, Vecteezy, and other open icon libraries (individual authors not retained).

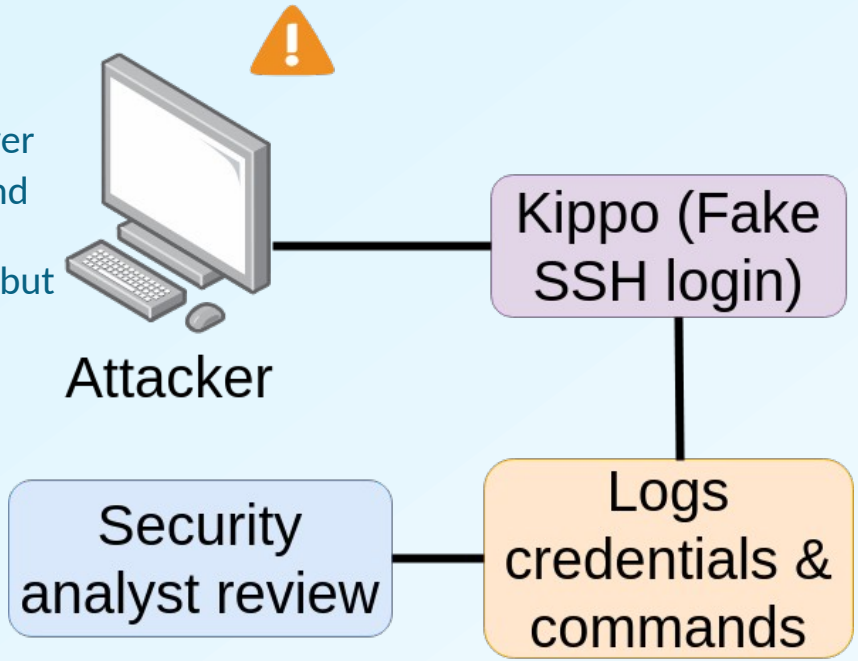
16.Microsoft Azure icons. Icons sourced from official Microsoft Azure icon sets, © Microsoft Corporation, used in accordance with their guidelines.

Contents



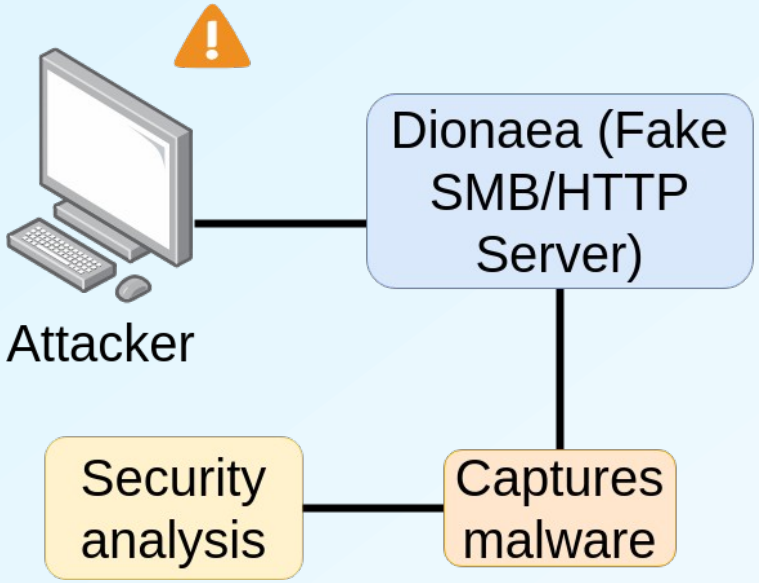
Example 1 – Kippo (SSH Honeypot)

- What is Kippo : A low-interaction SSH honeypot designed to detect and log brute-force attacks.
- How it works:
  - Attackers try logging into an emulated SSH server
  - Kippo records every login attempt and command entered
  - Attackers believe they are inside a real system, but everything is being monitored and stored
- Key Features:
  - Logs stolen credentials used by attackers
  - Can fake file system interactions, allowing attackers to explore but never cause real damage
  - Useful for analyzing attack techniques and identifying malicious IP addresses



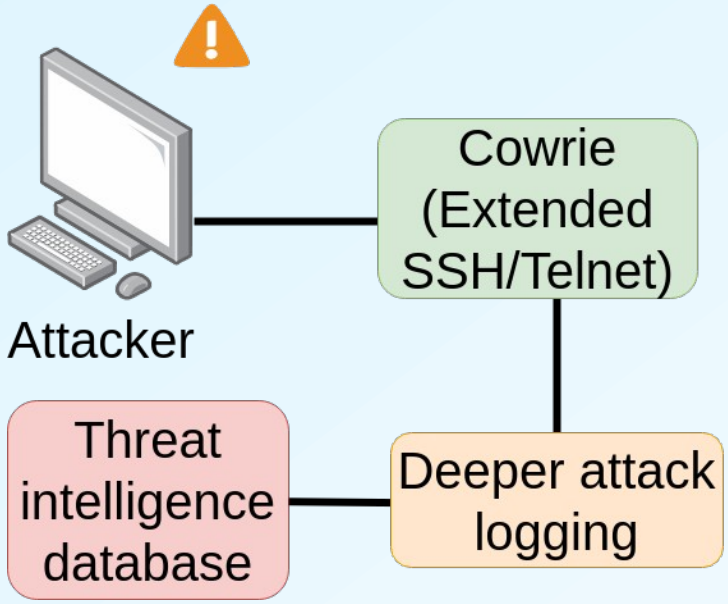
Example 2 – Dionaea (Malware Honeypot)

- What is Dionaea : A malware-catching honeypot that simulates vulnerable services to attract attackers.
- How it works:
  - Attackers target fake vulnerabilities in services like SMB, HTTP, and FTP
  - Dionaea captures and logs malicious payloads, including malware samples
  - It analyzes network traffic and helps security teams identify attack origins
- Key Features:
  - Detects and collects malware binaries for further study
  - Supports automatic malware analysis using external services
  - Helps identify botnet command & control (C2) traffic

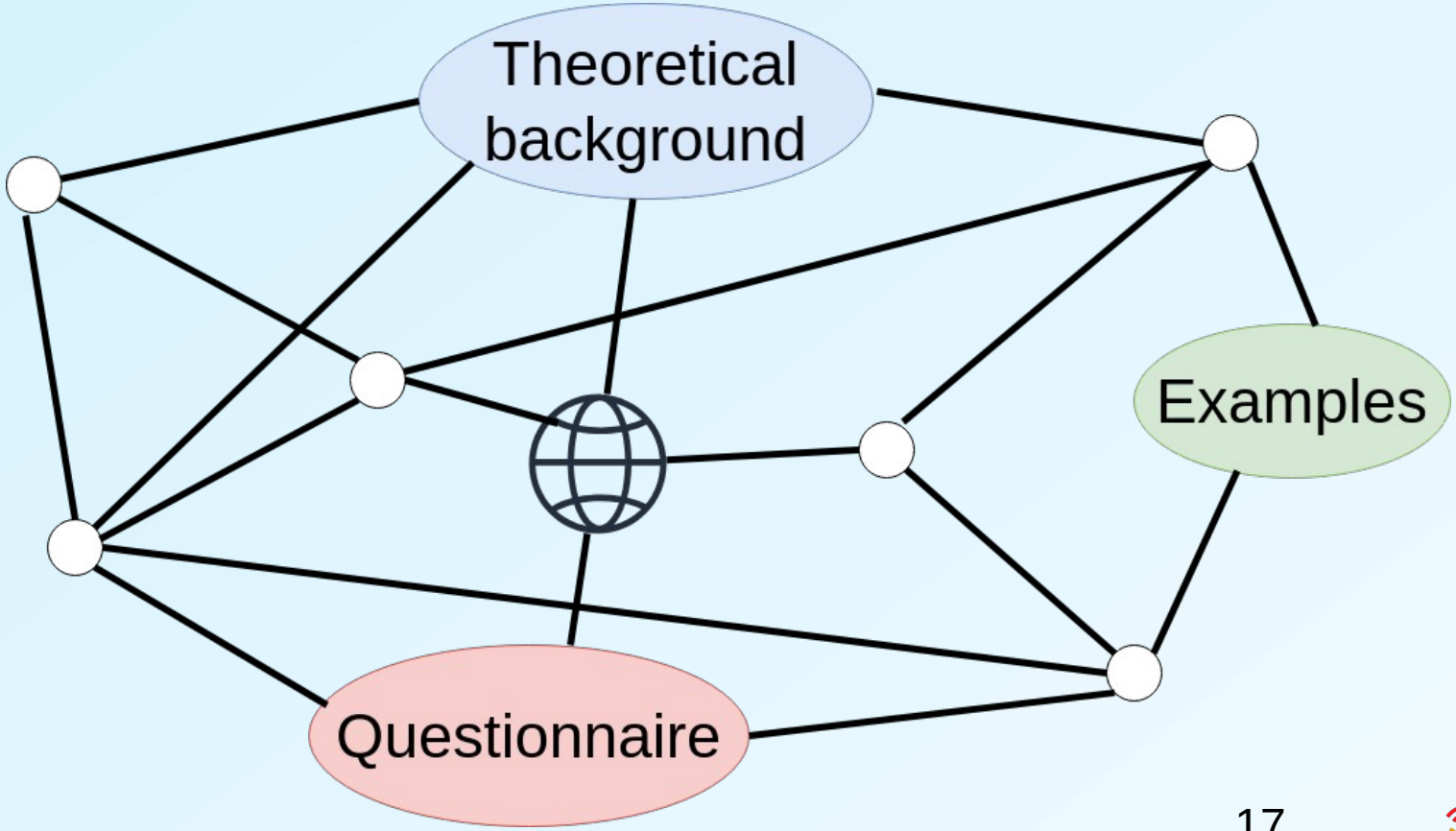


Example 3 – Cowrie (Extended SSH/Telnet Honeypot)

- What is Cowrie : An enhanced SSH/Telnet honeypot, based on Kippo, but with more advanced features.
- How it works:
  - Attackers attempt to brute-force SSH or Telnet
  - Cowrie records login attempts and attacker commands
  - Unlike Kippo, Cowrie allows realistic file interactions (e.g., fake downloads, system commands)
- Key Features:
  - Logs brute-force attempts and attack scripts
  - Can simulate real systems, making it harder for attackers to detect
  - Used in real-world research to analyze botnets, credential stuffing, and cybercrime trends



Contents



Question 1

What is the primary goal of a honeypot?

i. To block cyber attacks

ii. To collect intelligence on attackers

iii. To replace firewalls

iv. To train employees

Question 1

What is the primary goal of a honeypot?

Explanation: A honeypot is designed to attract attackers and observe their behavior, not to directly block them. Its main value lies in logging actions, techniques, payloads, and attack patterns that can later improve defenses.

i. To block cyber attacks (× Blocking is mainly handled by firewalls and IDS/IPS, not honeypots)

ii. To collect intelligence on attackers (✓Correct: honeypots gather detailed data on attacker behavior)

iii. To replace firewalls (×Honeypots complement, not replace, traditional security tools)

iv. To train employees (×Training is indirect, not the honeypot's core purpose)

Question 2

Which type of honeypot provides the most detailed attacker interaction?

i. Low-interaction
ii. High-interaction
iii. Medium-interaction
iv. IDS-based

Question 2

Which type of honeypot provides the most detailed attacker interaction?

Explanation: High-interaction honeypots simulate real production systems, allowing attackers to fully interact with services, file systems, and processes, providing richer intelligence.

i. Low-interaction (× Limited simulation, fewer insights)
ii. High-interaction (✓Correct: Full system behavior and deeper attacker engagement)
iii. Medium-interaction (× Not defined in your course material)
iv. IDS-based (× IDS is not a honeypot type)

Question 3

What is a key risk of deploying honeypots?

i. They increase network speed

ii. They can be used as an attack pivot

iii. They always block threats

iv. They replace antivirus software

Question 3

What is a key risk of deploying honeypots?

Explanation: If not properly isolated, attackers may exploit a honeypot as a stepping stone (pivot) to reach real internal systems.

i. They increase network speed (× Honeypots do not affect performance positively)

ii. They can be used as an attack pivot (✓Correct: poor isolation creates risk)

iii. They always block threats (× Honeypots observe rather than block)

iv. They replace antivirus software (× Honeypots complement security layers, they do not replace them)

Question 4

Which of the following is NOT a real honeypot?

i. Dionaea

ii. Snort

iii. Kippo

iv. Cowrie

Which of the following is NOT a real honeypot?

Explanation: Snort is an Intrusion Detection/Prevention System (IDS/IPS), not a honeypot. It monitors and analyzes traffic but does not act as a decoy system.

i. Dionaea (✓Correct: Malware honeypot)

ii. Snort (× IDS/IPS, not a honeypot)

iii. Kippo (✓Correct: SSH honeypot)

iv. Cowrie (✓Correct: Advanced SSH/Telnet honeypot)

Question 5

How should honeypots be placed in a network?

i. Directly in the internal network
ii. Isolated or in the DMZ
iii. Connected to the core database
iv. Replacing the firewall

How should honeypots be placed in a network?

Explanation: Honeypots must be isolated so attackers cannot reach real assets. Placement in the DMZ allows exposure while protecting internal systems.

i. Directly in the internal network (× Dangerous and risky)

ii. Isolated or in the DMZ (✓Correct: Correct and safe deployment)

iii. Connected to the core database (× Creates critical exposure)

iv. Replacing the firewall (× Honeypots complement, not replace, firewalls)