

Forum:[Hack Forums](#)

Thread: [Complete Guide to Identifying and Verifying Honeypots](#)

Chorozon - March 5, 2025, 7:15 am

Hello everyone,

Today, I want to dive into a fascinating topic that every cybersecurity enthusiast, ethical hacker, or red teamer should understand: honeypots. These decoy systems are designed to mimic legitimate targets, luring attackers in order to study their tactics, waste their time, and gather valuable threat intelligence. However, identifying and verifying honeypots is equally important—whether you're defending your network or researching attack patterns.

In this guide, I'll walk you through everything you need to know about honeypots, including how they work, how to identify them, and tools/techniques to verify their presence. Let's get started!

What Are Honeypots?

Honeypots are traps set up by defenders to deceive attackers. They can range from simple fake services to fully functional systems designed to mimic real-world environments. Their primary purposes include:

- Detecting Attacks : Identifying malicious activity early.

- Gathering Intelligence : Learning attacker tactics, techniques, and procedures (TTPs).

- Wasting Resources : Distracting attackers from real targets.

There are two main types of honeypots:

- Low-Interaction Honeypots : Simulate basic services (e.g., SSH, FTP) without exposing a full system. Examples include Cowrie (SSH honeypot) and Dionaea (malware honeypot).

- High-Interaction Honeypots : Fully functional systems that allow attackers to interact deeply, often used for advanced research. Tools like Honeyd fall into this category.

Red Flags That Point to a Honeypot

Before diving into technical tools, let's talk about behavioral and environmental indicators that might suggest you're dealing with a honeypot. These red flags are critical for both attackers and defenders to recognize:

1. Lack of Legitimate Activity

- No User Accounts or Files : If the system has no real user accounts, files, or applications beyond default configurations, it's likely a honeypot.

- No Outbound Traffic : Honeypots rarely send outbound traffic (e.g., emails, updates, or API calls).

2. Overly Permissive Access

- Weak Credentials : Honeypots often use obvious credentials like admin:admin or root:root to lure

attackers.

Instant Success : If your first attempt at exploitation works flawlessly, be suspicious—it could be too good to be true.

3. Unusual Services

Non-Standard Ports : For example, an SSH service running on port 8080 instead of the standard port 22.

Outdated Software : Honeypots may intentionally run outdated versions of software to attract attackers.

4. Instant Logging/Alerting

Immediate Response : Honeypots are often configured to log every interaction immediately. For instance, scanning a port might trigger alerts or responses within seconds.

5. Fake Sensitive Files

Lure Files : Honeypots may contain fake "sensitive" files like passwords.txt, credentials.db, or confidential.pdf. These files are designed to entice attackers but serve no real purpose.

6. Unrealistic System Behavior

No Real Users : If you interact with a system and notice no signs of real users (e.g., no browser history, no recent documents), it's a strong indicator of a honeypot.

Limited Functionality : High-interaction honeypots may allow some commands but fail when executing complex tasks (e.g., compiling code or installing packages).

7. Suspicious Metadata

WHOIS Records : Check the IP address using WHOIS lookup tools. If the IP is registered to a security company or labeled as a honeypot, proceed with caution.

Shodan/Censys Tags : Search the IP on platforms like Shodan or Censys. Tags like "honeypot" or "cowrie" are dead giveaways.

8. No DNS Queries

No Internet Activity : Honeypots often don't perform DNS queries or communicate with external servers. Use tools like Wireshark to monitor traffic—if there's no outbound DNS traffic, it's suspicious.

9. Default Configurations

Default Banners : Many honeypots use default service banners that are easy to fingerprint. For example:

Cowrie SSH honeypot uses SSH-2.0-OpenSSH_6.0p1 Debian-4+deb7u2.

Dionaea malware honeypot often exposes SMB shares with default names.

10. Unrealistic Network Topology

Isolated Systems : Honeypots are often standalone systems with no connection to other devices on the network. Use tools like Nmap to scan the subnet—if the target is isolated, it could be a honeypot.

How to Identify and Verify Honeypots

Now that you know what to look for, let's explore the technical methods and tools you can use to confirm whether a system is a honeypot.

1. Network Scanning

Start by performing a network scan using tools like Nmap to gather information about the target system:

Code `nmap -O -sV -p- <IP>`

Look for Anomalies :

Mismatched OS/service versions (e.g., Windows OS with Linux SSH banners).

Default banners from known honeypot software (e.g., Cowrie's SSH banner: SSH-2.0-OpenSSH_6.0p1 Debian-4+deb7u2).

2. Traffic Analysis

Use tools like Wireshark or tcpdump to analyze network traffic:

Red Flags :

No DNS queries or legitimate user traffic.

Traffic only to/from known honeypot IPs (check databases like GreyNoise or Shodan).

3. System Interaction

Interact with the system to test its behavior:

Test Credentials : Try default passwords like admin:admin. If accepted, it's likely a honeypot.

Upload/Download Files : Honeypots often lack real file systems. Attempting to upload or download files may fail or behave strangely.

4. Reverse DNS and WHOIS Lookup

Check the IP address for ownership details:

Code `whois <IP>` Red Flags :

IPs registered to security firms (e.g., Rapid7, AlienVault).

Tags like "honeypot" in WHOIS records or on platforms like Shodan

Tools for Detecting Honeypots

Here are some specialized tools and platforms to help you identify honeypots:

1. Honeyscore

Purpose : Scans IPs for honeypot indicators like open ports, service anomalies, and metadata.

Website : <https://honeyscore.io>

2. GreyNoise

Purpose : Filters out known honeypot traffic from internet-wide scans.

Website : <https://greynoise.io>

3. Shodan/Censys

Purpose : Search for tags like “honeypot” or “cowrie” in IP metadata.

Example Query :

Code product:"Cowrie SSH Honeypot"

4. Honeyd Detect
Purpose : Identifies low-interaction honeypots by analyzing TCP/IP stack inconsistencies.

Pr0m3th3us - March 5, 2025, 7:19 am

Thanks for the interesting read.

Chorozone - March 5, 2025, 7:21 am

(Mar 5, 2025 08:19 AM)SIM-2567248 Wrote: [\[CITE\]](#) Thanks for the interesting read.

Thanks man

Feel free to add anything