**How to Build a Network Honeypot**

**By Adel Dakhlallah**

**Goal**: Create a basic honeypot that mimics a vulnerable service to log and analyze attack behavior. Showcase your skills in cybersecurity, logging, networking, and scripting.

**Step 1: Choose Your Honeypot Type**

Decide what kind of honeypot you want to build:

* **Low-interaction honeypots** (safer, easier):
  + Tools: Cowrie (SSH/Telnet), Honeyd, Dionaea, Glastopf (HTTP)
* **High-interaction honeypots** (real services, more risk):
  + Tools: Build a vulnerable VM (e.g., Metasploitable or custom setup)

**Step 2: Set Up Your Environment**

Use a safe, isolated system or cloud VM (e.g., AWS, DigitalOcean, VirtualBox):

* Install Ubuntu/Debian
* Update system with following command:
  + sudo apt update && sudo apt upgrade
* Install Git, Python, and dependencies

**Step 3: Install the Honeypot**

**Example: Install Cowrie**

* Install dependencies:
  + sudo apt install git python3-venv python3-pip libssl-dev libffi-dev build-essential
* Clone Cowrie repo
  + git clone <https://github.com/cowrie/cowrie.git> cd cowrie
* Setup virtual environment
  + python3 -m venv cowrie-env

source cowrie-env/bin/activate

* Install Python dependencies
  + pip install --upgrade pip
  + pip install -r requirements.txt
* Copy config template
  + cp etc/cowrie.cfg.dist etc/cowrie.cfg
* Configure as needed in etc/cowrie.cfg.

**Step 4: Run the Honeypot**

* Command:
  + bin/cowrie start
* Cowrie simulates an SSH/Telnet service
* It logs credentials, commands, and sessions attackers try
* Logs are stored in var/log/cowrie

**Step 5: Monitor and Analyze**

* Use Kibana/Elasticsearch or custom scripts to analyze logs
* Look for:
  + Brute force attempts
  + Common malware payloads
  + Attacker IPs and behaviors

**Step 6: Add Security Safeguards**

* Use firewalls or sandboxing (e.g., run in a container or isolated VM)
* Do **not expose it on your home network** unless secured
* Use a VPS with specific ports open