**Project:** T-Pot Honeypot: Real-Time Attack Detection in Azure

Wilton Lizardo

**Goal**

This project explores the use of T-Pot, a multi-honeypot platform, deployed in Microsoft Azure to collect and analyze real-world cyberattacks. Honeypots act as decoy systems that attract attackers, allowing us to monitor their techniques in a controlled environment.

Using T-Pot’s integrated ELK stack (Elasticsearch, Logstash, Kibana), we captured attack data, visualized trends, and gained insights into brute force attempts, port scans, and global threat activity.  
The goal was to develop hands-on cybersecurity skills in cloud deployment, network defense, and threat intelligence.

**Methodology**

Environment Setup

Platform: Microsoft Azure

Instance Type: Ubuntu 22.04 LTS (4 vCPUs, 16GB RAM, 256GB SSD)

Honeypot System: T-Pot 24.04.1 (multi-honeypot framework)

Data Collection Approach

Allowed unrestricted inbound traffic to simulate a vulnerable system.

Captured attack data in Kibana, Suricata IDS, and Elastic Stack.

Logged brute-force attempts, vulnerability scans, and exploit attempts.

Attack Monitoring & Analysis

Monitored traffic over 15 hours to study attack frequency and trends.

Analyzed attacker origins, targeted ports, brute-force credentials, and CVEs exploited. Classified threats based on honeypot services (SSH, SMB, SIP, RDP, etc.).

**Step 1**

I set up an Ubuntu-based Virtual Machine on Azure with the following specifications:

VM Name: honeyp

Resource Group: HONEYPOT-RG

 Region: East US

Operating System: Ubuntu (Linux)

VM Size: Standard\_D4s\_v3 (4 vCPUs, 16GB RAM)

Disk Space: 256GB

**Step 2: Configure Inbound Network Security Rules**

After setting up the VM, I opened the required ports in the Network Security Group (NSG). Security Group Configuration (As Seen in Screenshot)

SSH (Port 22, TCP) → Allowed (open to myip for remote access)

Custom Rule (Ports 1-65535) → Allowed

Since this was a test environment, I opened all ports (1-65535) inbound to capture a wide range of attacks across multiple services. For production environments, please limit opened ports only to required ones.

A screenshot of a computer

AI-generated content may be incorrect.

**Step 3: T-pot installation**

This is the GitHub repository for T-Pot installation. You can access the link by following the installation guide.

https://github.com/telekom-security/tpotce

A screenshot of a computer

AI-generated content may be incorrect.

T-Pot is a multi-honeypot platform that integrates over 20+ honeypots into a single system. It provides live attack visualizations, logs using the Elastic Stack, and supports multiple architectures (AMD64, ARM64).

The installation is straightforward, requiring 8-16GB RAM and 128GB disk space. The installation command uses a simple bash script to set up the environment.

A white background with black and white clouds

AI-generated content may be incorrect.

T-Pot supports different environments: VMs, Cloud, Hardware, and Raspberry Pi. I did my deployment in the cloud.

**Step 1: Update and Prepare the System**

1 Open Putty in your desktop

2 Connect Putty to Ubuntu server to Update and Prepare the system also to write scripts

3 In “Host name (or IP address)” type the VM public Ip address with the “Port” set to 22.

Select “Connection type:” to “SSH” then click open

A screenshot of a computer

AI-generated content may be incorrect.

**A screenshot of a computer screen

AI-generated content may be incorrect.**

Run script (sudo apt update) to update package lists.

The system fetched updated package information from Ubuntu repositories, ensuring the latest security patches and dependencies.

**Step 2: Clone the T-Pot Repository**

**A screen shot of a computer

AI-generated content may be incorrect.**(Cloning the Repository)

**Step 3: Install T-Pot**

**A screenshot of a computer

AI-generated content may be incorrect.**

Navigated into the cloned directory **(cd tpotce)** and started the installation script with this command- **(./install.sh).** The script began installing dependencies, including: python3, ansible e.t.c.

A screenshot of a computer

AI-generated content may be incorrect.

Ansible is an automation tool used for configuring and deploying systems, and in this case, it's automating the setup of T-Pot by installing dependencies, configuring settings, and setting up Docker-based honeypots.

**Step 3: Choose T-Pot Installation Type: I Chose Hive A screenshot of a computer program

AI-generated content may be incorrect.A black screen with white text

AI-generated content may be incorrect.**I typed H to choose

**Step 4: Configure T-Pot User**

**A black screen with white text

AI-generated content may be incorrect.**(User Setup)

Entered a username (use any name you like).

Set up a password for accessing the Web UI.

**Step 5: Downloading Honeypot Images**

**A screenshot of a computer

AI-generated content may be incorrect.**

The system started pulling Docker-based honeypots. These images are containers used by T-Pot to simulate vulnerable systems.

**Step 6: Restarted the VM: Go to your virtual machine and restart it.**

**Step 4: Accessing the T-Pot Web Interface**

**A screenshot of a computer

AI-generated content may be incorrect.**The T-Pot web dashboard was accessed using: https://<VM-IP>:64297

A login prompt appeared, requiring the username and password that were set during installation.

The browser displayed a certificate warning because T-Pot uses a self-signed SSL certificate by default.

This is normal for self-hosted applications.

The warning can be bypassed by clicking “Visit this website”.

**Step 1: Exploring the T-Pot Dashboard**

A screenshot of a computer

AI-generated content may be incorrect.

T-Pot Web Interface Main Page

After logging in, the T-Pot 24.04.1 dashboard was displayed.

The dashboard provides access to several honeypot tools:

Attack Map – Visual representation of detected attacks in real time.

CyberChef – A tool for data manipulation, encoding, and forensic analysis. Elasticvue – A web UI for interacting with Elasticsearch logs.

Kibana – A powerful dashboard for visualizing honeypot activity.

Spiderfoot – An OSINT tool for gathering intelligence on attackers.

The timestamp (19:08, 11/03/2025) shows the current server time.

**Step 2: Viewing the Real-Time Attack Map**

A map of the world

AI-generated content may be incorrect.

Viewing the Real-Time Attack Map

Attack Map with Live Honeypot Data

The Attack Map is a real-time visualization of incoming attacks from dierent locations. The table at the bottom displays:

Attack Service: (e.g., FTP, SSH, Telnet, Email, SQL, DNS)

Hits: Number of recorded attacks.

Source IP Address: The IP from which the attack originated. I covered the ip adresses, ideally, you can see them.

Country: The geographical location of the attacker.

Timestamp: When the attack was detected.

Honeypot Service: Which honeypot instance detected the attack.

**Analysis of the T-Pot Kibana Dashboard**

As of the time of this documentation, T-Pot has been running for approximately 5 hours and 30 minutes.

**Screenshot 1**



**Honeypot Attacks Summary (Top)**

Total attacks recorded: 26,000

The top attacking services include:

Cowrie (SSH Honeypot) → 12,000 attacks (Attackers target SSH for brute-force attacks to gain remote access to Linux servers.)

Honeytrap → 7,000 attacks (General-purpose honeypot that mimics vulnerable services, attracting automated bots and exploit kits.)

Sentrypeer → 1,000 attacks (Likely targeted for its ability to mimic open network services, making it an attractive deception honeypot for attackers.)

Dionaea (Malware Honeypot) → 255 attacks (Attracts malware payloads used in real-world attacks, making it a prime target for automated delivery systems.)

Other honeypots like Adbhoney, MiniPrint, ConPot, and RedisHoneypot have fewer but still significant attacks.

Tanner, Honeytrap, and Cowrie are the most attacked services.

**Honeypot Attacks Bar Graph (Bottom Left)**

This bar chart visually represents the attack volume per honeypot type.

Cowrie and Honeytrap have the highest attack counts, followed by Tanner. Less commonly attacked honeypots like ElasticPot, HoneyAML, and Discompt show minimal activity.

This helps in identifying which honeypots are most attractive to attackers.

**Honeypot Attacks Over Time (Bottom Middle)**

This histogram graph tracks:

Total attack activity (green line).

Unique attacking IPs (orange line).

There was a surge in attacks around May 7, 2025, at 1:00am.

The attack volume continues to fluctuate, with peaks and dips over time.

Shows attack trends, helping to detect periods of high malicious activity.

**Attack Map – Dynamic (Bottom Right)**

This world map visualization displays attack origins and aected locations. Provides a geographic view of attack sources and targeted regions.

**Screenshot 2**

A screen shot of a computer

AI-generated content may be incorrect.

**Attacks by Destination Port Histogram (Left Chart)**

This chart visualizes attacks grouped by targeted ports.

The most attacked ports:

50995 (Lime Green) → Mostly to find Custom Services, Misconfigured services, Backdoors.

58000 (Green) → Custom apps, VPNs, Web shells, RATs (Remote Access Trojans)

20256(Orange) → Bots and attackers often scan all ports looking for any response.

81 → Scanning for vulnerable devices (home routers, DVRs and IP cameras).

**Attacks by Country Histogram (Right Chart)**

This chart categorizes attacks by country.

Top attacking countries:

United States (Green)

United Kingdom (Purple)

Brazil (turquoise)

Hong Kong (Pink)

The Netherlands (Light Yellow)

The United States had consistent attack activity, while United Kingdom and Brazil had multiple spikes in attack volume.

Conclusion: United Kingdom and the U.S. are major attack sources, followed by Brazil and Hong Kong.

**Screenshot 4**

A screenshot of a computer

AI-generated content may be incorrect.

**Attacker SI-CP IP Reputation (Left Chart)**

This chart categorizes attackers based on reputation:

Orange (Known Attacker) → Majority of attacks originate from previously flagged malicious IPs.

Blue (Mass Scanner) → A portion of attackers are using mass scanning tools like Shodan, Censys, or masscan.

Cyan (Bot, Crawler) → A small percentage of attacks come from web crawlers or automated bots.

**Conclusion: Most attacks originate from known malicious sources rather than random scans.**

**Attacks by Honeypot (Second Chart)**

This chart distributes attacks across different honeypots.

The most attacked honeypots are:

Honeytrap (Red) → Highest attack volume.

Dionaea (Green) → A deceptive honeypot.

Cowrie (Yellow) → Simulated SSH honeypot.

Adbhoney (Pink) → Malware collection honeypot.

Lesser attacks are recorded for Sentrypeer, ConPot (SCADA honeypot), and Tanner.

**Conclusion: Attackers primarily target SSH (Cowrie) and deception-based honeypots like Honeytrap and Dionaea.**

**Port OS Distribution (Third Chart)**

This chart shows fingerprinted OS versions from attackers.

The most common OS fingerprints:

Linux 2.2.x-3.x (Light Blue) → Majority of attacking hosts run Linux.

Windows NT kernal (Blue) → Some attacks originate from outdated Windows machines. Linux 3.11 and newer (Purple) → A portion of attacks come from more recent Linux systems.

**Conclusion: Most attacking systems are running Linux, with some Windows-based attackers. Attacks by Country (Right Chart)**

This chart categorizes attacks by source country.

Top attack origin countries:

United States (Green)

United Kingdom (Purple)

Brazil (Light Blue)

Hong Kong (Pink)

The Netherlands (Yellow)

Other sources include Canada, France, China, Germany, and Turkiye.

Conclusion: U.S and the U.K are the leading sources of attacks, followed by Brazil and Hong Kong.

**Screenshot 5**

A screenshot of a graph

AI-generated content may be incorrect.

**Attacks by Country and Port (Left Section)**

This set of donut charts shows how dierent countries are targeting specific ports. Top attacking countries:

United States (Orange) → Primarily attacking port 20256.

United Kingdom (Mix) → Attacks spread across multiple ports:

20256

81

2465

Brazil (Green) → Mostly attacking 81 .

Hong Kong (Green & Blue) → Targeting 1158 and other minor ports.

Netherlands (25% Multi-colored Chart) → A mix of ports.

**Suricata Alert Category Histogram (Right Section)**

A screenshot of a computer

AI-generated content may be incorrect.

This graph tracks real-time intrusion detection alerts captured by Suricata. Top attack categories:

Generic protocol Commands (30 alerts) → Indicates brute-force attacks or privilege escalation attempts.

Attempted Information Leak (40 alerts) → Attackers probing for sensitive data leaks. Surge in attack attempts around 01:00 AM, May 7, 2025.

The histogram shows a steady increase in attack frequency leading to large spikes in administrator privilege escalation attempts.

**Conclusion:**

There was a major attack event around 01:00 AM targeting admin access. Repeated privilege escalation attempts suggest active exploitation eorts. Attackers are probing for vulnerabilities across multiple ports, particularly on SIP, SSH, and MikroTik services.

**Screenshot 6**

A screenshot of a computer screen

AI-generated content may be incorrect.

**Username Tag Cloud (Left Chart)**

The most frequently attempted usernames are:

"root" (largest text) → Attackers overwhelmingly attempt to log in as the root user. "postgres" → Indicates attempts to compromise PostgreSQL databases.

"oracle" → Targeting Oracle database instances.

"admin" → Generic administrator username.

Other targeted accounts:

"docker" → Likely targeting misconfigured containerized environments.

"gitlab-runner" → Attackers trying to exploit CI/CD pipelines.

"minecraft" → Indicates potential gaming server brute-force attacks.

"nginx," "tomcat," "mysql," "zabbix," "rancher" → Common software services targeted. **Conclusion:**

Attackers primarily attempt to log in as "root," "postgres," and "admin."

There is a strong focus on cloud, database, and container services.

DevOps tools (GitLab, Docker, Rancher) are also being targeted.

**Password Tag Cloud (Right Chart)**

The most commonly attempted passwords include:

"123456" (largest text) → The most frequently tried password.

Other weak passwords:

"admin123," "password," "1234," "abc123," "root," "111111," "qwerty123."

Pattern-based passwords:

"1qaz2wsx" (keyboard pattern)

"!QAZ@WSX" (shifted keyboard pattern)

Security-focused words like "elastic," "docker," and "bigdata" indicate attackers trying to breach specific services.

**Screenshot 7**A screenshot of a phone

AI-generated content may be incorrect.

**Major sources of attacks:**

OVH SAS (ASN 16276) → 15,605 attacks

Google Cloud (ASN 396982) → 6,760 attacks

DigitalOcean (ASN 14061) → 5,671 attacks

ENTEL Chile (ASN 27651) → 5,585 attacks

GoDaddy, SuperData, Unmanaged Ltd → Indicate hosting providers being used.

This suggests that many attacks originate from cloud-based infrastructure (Google Cloud, OVH, DigitalOcean)—often used for botnets, automated scanning, or exploitation attempts.

**Screenshot 8**

**A screenshot of a computer

AI-generated content may be incorrect.**

Suricata CVE - Top 10 (Third Panel)

Shows the top exploited vulnerabilities (Common Vulnerabilities & Exposures - CVEs): CVE-2002-001 (35 detections)

CVE-2002-001 (21 detections)

CVE-2001-041 (11 detections)

CVE-2018-148 (8 detections)

CVE-2019-119 (8 detections)

CVE-1999-011 (2 detections)

**Report**

**1. The Internet is Constantly Scanned for Vulnerabilities**

The honeypot was online for only 6 hours and still received 26,000+ attacks. Most attacks were automated—from bots, scanners, and compromised cloud-based servers.

**Inference:**

Any system exposed to the internet will be targeted, whether real or not. Organizations must assume their public-facing services are under constant threat.

**2. Attackers Prioritize Commonly Exploited Services**

The honeypot showed a preference for attacks on:

50995 Mostly to find Custom Services, Misconfigured services, Backdoors.

58000 Custom apps, VPNs, Web shells, RATs (Remote Access Trojans)

20256 Bots and attackers often scan all ports looking for any response.

81 Scanning for vulnerable devices (home routers, DVRs and IP cameras).

**Inference:**

These services are the biggest attack surfaces globally.

 If an organization doesn’t need a service (like SIP or RDP), it should be blocked.  Exposed SSH must use key-based authentication, not passwords.

**3. Cloud Hosting Providers Are Being Used for Attacks**

Major attacking ASNs came from:

OVH SAS, Google Cloud, DigitalOcean, GoDaddy.

**Inference:**

Attackers leverage cloud providers to launch automated attacks.

Blocking or monitoring trac from these sources is crucial.

Enterprises should apply rate limiting, geo-blocking, or threat intelligence feeds.

**4. Credential Attacks Are Still a Major Problem**

Most attempted usernames: "root", "admin", "postgres", "docker".

Most attempted passwords: "123456", "password", "admin123".

**Inference:**

Attackers assume many organizations still use default or weak credentials. Credential stung is a low-effort, high-reward attack method.

Organizations must enforce strong password policies and MFA.

**5. Attackers Are Still Exploiting Old CVEs**

The honeypot logged exploitation attempts for CVEs as old as 2002.

The honeypot detected active exploitation of:

CVE-2002-001, CVE-2001-041, CVE-2019-119, CVE-2018-148.

**Inference:**

Many organizations still run unpatched legacy systems.

Attackers assume there will always be unpatched machines in the wild. Regular patching and vulnerability management are critical.

**6. Most Attacks Were Likely Not Targeted, But Opportunistic**

The honeypot received attacks immediately after deployment.

Attack trac came from multiple regions and cloud providers.

**Inference:**

Most attacks on the internet are automated, random, and not targeted at a specific company.

Attackers scan the internet looking for any vulnerable system they can exploit. Even a small or unknown organization is not "safe" from attack.

**7. Suricata IDS Proved Eective in Detecting Known Exploits**

The honeypot successfully flagged:

Nmap scans (reconnaissance).

SMB Exploits (EternalBlue).

DoublePulsar backdoor attempts.

**Inference:**

Intrusion Detection Systems (IDS) like Suricata are essential for identifying active threats. Organizations should integrate IDS alerts with automated blocking tools (e.g., fail2ban, SIEMs).

**Conclusion & Security**

**Recommendations**

This project successfully deployed an internet-facing honeypot using T-Pot on Azure to collect real-world threat intelligence. Over 15 hours, the honeypot recorded 70,000+ attack attempts, demonstrating that any publicly exposed system is continuously scanned and targeted by automated and manual attacks.

**Key findings from the project include:**

Critical remote services (SSH, SMB, SIP, and RDP) were the primary attack targets. Credential stung attacks used weak and default usernames/passwords. Old vulnerabilities (e.g., CVE-2002-001, CVE-2019-119) were still actively exploited. Cloud-hosted infrastructure (OVH, Google Cloud, DigitalOcean) was a major attack source. A significant number of attacks originated from known malicious ASNs, likely botnets.

These insights highlight the persistent and evolving nature of cybersecurity threats, reinforcing the importance of continuous monitoring, proactive security defenses, and threat intelligence analysis.

Security Recommendations

Reduce Public Exposure of Critical Services

Restrict access to SSH (22), SMB (445), RDP (3389), and SIP (5060) unless necessary. Use a VPN or Bastion Host instead of exposing remote access services directly. Enforce Network Security Groups (NSGs) or firewalls to allow only trusted IPs.

Strengthen Authentication & Credential Security

Disable password-based SSH login, enforce SSH key authentication.

Implement Multi-Factor Authentication (MFA) for all remote access services. Use strong, unique passwords and rotate credentials regularly.

Implement Intrusion Detection & Response

Deploy Fail2Ban or Suricata IDS to detect and block brute-force attacks.

Forward logs to a SIEM (Azure Sentinel, Splunk, Wazuh) for real-time analysis. Set up alerting rules for suspicious login attempts and exploit scans.

Patch & Harden Systems Against Exploited Vulnerabilities

Regularly apply security patches to mitigate known CVEs.

Disable SMBv1 and enforce SMB signing to prevent EternalBlue exploits.

Harden web services against VoIP fraud, SIP abuse, and RDP brute-force attacks. Use Threat Intelligence to Block Known Malicious Actors

Block known malicious ASNs and IPs from the honeypot findings.

Leverage Threat Intelligence Feeds to prevent access from high-risk networks. Use Geo-blocking or rate-limiting to reduce automated attacks.

**Final Words**

This documentation provides an in-depth analysis of the deployment, monitoring, and findings from my T-Pot honeypot project. Through this research, I was able to observe real-world cyber threats, analyze attacker behaviors, and gain insights into how organizations