

# Honeypot (T-POD)

## Parte 1: Introducción a los Honeypots

1. **Definición de Honeypots**
2. **Tipos de Honeypots**
  - **Honeypots de Baja Interacción:** Breve descripción y ejemplos.
  - **Honeypots de Alta Interacción:** Breve descripción y ejemplos.
3. **Componentes y Funcionamiento**
4. **Ventajas y Desventajas**
5. **Casos de Uso**

## Parte 2: Configuración de un Honeypot (T-POT)

1. **Introducción a T-POT**
2. **Requisitos Previos**
3. **Instalación de T-POT**
4. **Configuración de T-POT**
5. **Pruebas Iniciales**

## Parte 3: Análisis de Datos y Conclusiones

1. **Recolección de Datos**
2. **Análisis de los Datos Recopilados**
3. **Interpretación de los Resultados**
4. **Conclusiones Finales**
5. **Limitaciones y Trabajo Futuro**

## Consideraciones Finales

- **Referencias**
- **Apéndices**
- **Formato y Presentación**

## Paso a Paso Detallado

### Parte 1: Introducción a los Honeypots

1. **Investigación Inicial**
2. **Redacción**

### Parte 2: Configuración de T-POT

1. **Preparativos**
2. **Documentación del Proceso**

### 3. Escritura

## Parte 3: Análisis de Datos y Conclusiones

1. Recolección de Datos
2. Análisis:
3. Escritura

## Parte 1: Introducción a los Honeypots

### Definición de Honeypot

Un honeypot es una herramienta de seguridad que actúa como cebo o señuelo para ciberatacantes. Se utiliza para monitorizar ataques y aprender para poder evitarlos en el futuro. Recibirán los ataques antes que los sistemas críticos.

Los honeypot crean servicios falsos propensos a ser atacados, como un servidor web o una base de datos. Estos ataques se recogen y analizan, permitiendo obtener información de los ataques y su procedencia y preparar los sistemas reales ante posibles amenazas similares.

### Tipos de Honeypots

#### Honeypots de Baja Interacción

- Emulan algunos servicios y funcionalidades básicas del sistema
- Recopilan información básica y algo limitada del atacante con pocos recursos.
- Son fáciles de implementar y mantener.
- Útiles para detectar los primeros pasos de un ataque.

#### Honeypots de Alta Interacción

- Emulan un abanico muy amplio de servicios y comportamientos de un sistema real.
- Permiten realizar una investigación más profunda del ataque e identificar el nivel de riesgo de la amenaza, métodos y objetivo del atacante.
- Son más complejos y requieren más recursos y mantenimiento.

### Componentes y Funcionamiento

Un honeypot está compuesto por varios elementos esenciales que le permiten atraer y registrar las actividades de los atacantes. Estos componentes incluyen:

- **Red y Sistema Operativo:** El honeypot puede estar basado en una máquina virtual o física que imita un sistema operativo específico.
- **Servicios Falsos:** Se configuran servicios que parecen vulnerables, como servidores web, SSH, FTP, entre otros, que los atacantes suelen buscar.
- **Sistemas de Monitoreo:** Herramientas que registran todas las actividades del atacante, incluyendo intentos de explotación, comandos ejecutados y movimientos dentro del sistema.

El funcionamiento básico de un honeypot implica atraer al atacante hacia el sistema falso y registrar toda la actividad. La información recopilada se analiza posteriormente para entender mejor las tácticas y herramientas utilizadas por los atacantes, lo que ayuda a mejorar las defensas de la red real.

## Ventajas y Desventajas

### Ventajas

- **Detección Temprana de Ataques:** Los honeypots pueden identificar y alertar sobre intentos de intrusión en etapas tempranas.
- **Análisis Detallado y accesible:** Proporcionan una visión detallada de las tácticas y técnicas de los atacantes.
- **Educación y Entrenamiento:** Son útiles para entrenar al personal de seguridad en la identificación y respuesta a incidentes.
- **No requieren muchos recursos**

### Desventajas

- **Recursos Necesarios:** Los honeypots de alta interacción requieren un mantenimiento elevado.
- **Falsos Positivos:** Los honeypots pueden generar una gran cantidad de datos, incluyendo actividades no maliciosas, lo que complica el análisis.
- **Vulnerabilidades:** Los honeypots poseen las mismas vulnerabilidades que los sistemas que emulan, por lo que necesitan un mantenimiento constante para ser efectivos.
- **Configuración:** Si no se configuran correctamente, pueden ser utilizados como punto de partida para ataques al resto de la red.

## Casos de Uso

Los honeypots se utilizan en diversos contextos y escenarios para mejorar la seguridad informática:

**Investigación de Amenazas (Threat Intelligence)** Los honeypots se utilizan ampliamente en la investigación de amenazas para obtener información valiosa sobre nuevas amenazas y vulnerabilidades. Al observar directamente los métodos de los atacantes, los analistas de ciberseguridad pueden identificar patrones y desarrollar firmas de detección para nuevas variantes de malware. Esto pertenece a la rama de **Threat Intelligence**, que se centra en comprender y mitigar las amenazas a la seguridad a través de la recopilación y el análisis de información sobre posibles atacantes y sus métodos.

**Entornos Empresariales (Security Operations)** En las empresas, los honeypots se implementan para detectar y mitigar ataques internos y externos. Al integrar honeypots en la infraestructura de red, las organizaciones pueden identificar comportamientos sospechosos que podrían indicar una brecha de seguridad. Esto es especialmente útil para detectar ataques dirigidos y amenazas persistentes avanzadas (APT). Este caso de uso se clasifica dentro de **Security Operations**, que se encarga de la gestión diaria de la seguridad de una organización, incluyendo la detección y respuesta a incidentes.

**Educación y Capacitación (Cybersecurity Training)** Los honeypots se utilizan en entornos académicos y profesionales para enseñar ciberseguridad y para simular ataques reales durante entrenamientos. Proporcionan un entorno seguro y controlado donde los estudiantes y profesionales pueden aprender sobre las técnicas de los atacantes y practicar la respuesta a incidentes. Esto pertenece a la rama de **Cybersecurity Training**, que se enfoca en la formación y desarrollo de habilidades en ciberseguridad.

**Protección de Infraestructuras Críticas (Critical Infrastructure Protection)** Los honeypots son implementados en infraestructuras críticas, como servicios financieros, de energía y telecomunicaciones, para monitorear y proteger contra ataques avanzados. En estos entornos, es crucial detectar y mitigar cualquier intento de intrusión rápidamente.

Carlos Gutierrez Torrejon  
Álvaro García De La Mata

keepcoding

para evitar interrupciones significativas en los servicios esenciales. Este caso de uso se clasifica dentro de **Critical Infrastructure Protection**, que se centra en la defensa de los sistemas y activos vitales para la seguridad nacional y el bienestar público.

**Desarrollo y Pruebas de Seguridad (Security Research and Development)** Los honeypots también se utilizan en el desarrollo y pruebas de nuevas tecnologías de seguridad. Proporcionan un entorno donde los investigadores pueden probar la efectividad de nuevas defensas sin arriesgar los sistemas de producción. Además, permiten a los desarrolladores observar cómo los atacantes interactúan con nuevas tecnologías y ajustar las medidas de seguridad en consecuencia. Este caso de uso pertenece a la rama de **Security Research and Development**, que se dedica a la innovación y mejora continua de las tecnologías de seguridad.

Parte 2: Configuración de un Honeypot (T-Pot)

Vamos a implementar la solución T-Pot en su versión estándar, esto incluye sensores y herramientas de búsqueda y análisis como Elasticsearch, Suricata o Kibana.

Se va a implementar en una máquina virtual en Azure que monta un Debian 11.

Entorno virtual

Especificaciones

Virtual machines

Default Directory

Create

Switch to classic

Reservations

Manage view

Refresh

Export to CSV

Open query

Assign tags

Start

Restart

Stop

Delete

Filter for any field...

Subscription equals all

Type equals all

Resource group equals all

Location equals all

Add filter

Showing 1 to 1 of 1 records.

No grouping

List view

Name	Type	Subscription	Resource group	Location	Status	Operating system	Size	Public IP address	Disks
tpot	Virtual machine	Azure subscription 1	TPOT	East Asia	Running	Linux	Standard_D2s_v3	23.102.235.8	2

tpot

Virtual machine

Search

ConnectStartRestartStopHibernateCaptureDeleteRefreshOpen in mobileFeedback

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Connect

Connect

Bastion

Networking

Advisor (1 of 2) Use Availability zones for better resiliency and availability

Essentials

Resource groupTPOT

StatusRunning

LocationEast Asia

SubscriptionAzure subscription 1

Subscription ID7b412b03-2348-4c34-b45f-e6d1fd7516ae

Operating systemLinux (debian 11)

SizeStandard D2s v3 (2 vcpus, 8 GiB memory)

Public IP address23.102.235.8

Virtual network/subnettpotKeepcoding-vnet/default

DNS nameNot configured

Health state-

Time created12/7/2024, 21:48 UTC

- Properties
- Monitoring
- Capabilities (7)
- Recommendations (2)
- Tutorials

Virtual machine		Networking	
Computer name	tpot	Public IP address	23.102.235.8 ( Network interface tpot436 )
Operating system	Linux (debian 11)	Public IP address (IPv6)	-
VM generation	V2	Private IP address	10.0.0.5
VM architecture	x64	Private IP address (IPv6)	-
Agent status	Ready	Virtual network/subnet	tpotKeepcoding-vnet/default
Agent version	2.2.47	DNS name	Configure
Hibernation	Disabled	Size	
Host group	-	Size	Standard D2s v3
Host	-	vCPUs	2
Proximity placement group	-	RAM	8 GiB

↻ Swap OS disk

Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (...)	Encryption ⓘ	Host caching ⓘ
<a href="#">tpot_OsDisk_1_5551aba7</a>	Standard HDD LRS	30	500	60	SSE with PMK	<a href="#">Read/write</a> ▼

Data disks

Showing 1 of 1 attached data disks

[+](#) Create and attach a new disk [↻](#) Attach existing disks

LUN ⓘ	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (...)	Encryption ⓘ
0	<a href="#">tpot_DataDisk_0</a>	Standard SSD LRS	128	500	100	SSE with PMK

## Reglas NSG

Se han implementado tres reglas NSG para permitir el acceso por SSH y WEB desde nuestras IPs públicas y permitir el tráfico desde todo internet a los puertos específicos que utiliza T-Pot

Rules [⌵](#) Collapse all

Network security group [tpot-nsg](#) (attached to networkInterface: [tpot436](#))  
Impacts 0 subnets, 1 network interfaces [+ Create port rule](#)

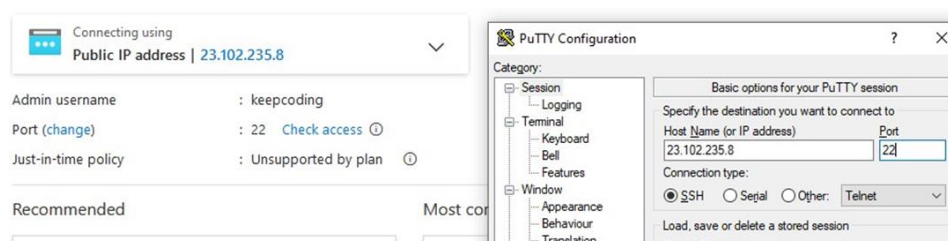
Source == all Destination == all Protocol == all Action == all

Priority ↑	Name	Port	Protocol	Source	Destination	Action
100	POT-SSH-IN	64295	TCP	207.188.138.51,83.55.106.149	Any	Allow
110	POT-WEB	64297	TCP	207.188.138.51,83.55.106.149	Any	Allow
120	TPOT-ALL	0-64293,64298-...	TCP	Any	Any	Allow
65000	AllowVnetInbound ⓘ	Any	Any	VirtualNetwork	VirtualNetwork	Allow

## Instalación T-Pot

<https://github.com/telekom-security/tpotce?tab=readme-ov-file#choose-your-distro>

→ Una vez montada la máquina virtual, nos conectamos por SSH



→ Actualizamos el sistema

```
sudo apt update
```

```
keepcoding@tpot:~$ sudo apt update
[sudo] password for keepcoding:
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://deb.debian.org/debian bullseye-updates InRelease
Hit:3 http://security.debian.org/debian-security bullseye-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
```

→ Instalamos curl

```
sudo apt install curl
```

```
keepcoding@tpot:~$ sudo apt install curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcurl4
The following NEW packages will be installed:
  curl libcurl4
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
```

→Ejecutamos el instalador sin permisos root

```
env bash -c "$(curl -sL https://github.com/telekom-  
security/tpotce/raw/master/install.sh)"
```

```
keepcoding@tpot:~$ env bash -c "$(curl -sL https://github.com/telekom-security/tpotce/raw/master/install.sh)"
Co
Pu
T-Pot Installer
err
ng### This script will now install T-Pot and all of its dependencies.
ng### Install? (y/n) █
k:
```

→ Seleccionamos la instalación estándar, que incluye todo lo necesario (sensores, elastic, kibana, etc)

```
### Choose your T-Pot type:
### (H)ive - T-Pot Standard / HIVE installation.
###      Includes also everything you need for a distributed setup with sensors.
### (S)ensor - T-Pot Sensor installation.
###      Optimized for a distributed installation, without WebUI, Elasticsearch and Kibana.
y:### (M)obile - T-Pot Mobile installation.
###      Includes everything to run T-Pot Mobile (available separately).
### Install Type? (h/s/m) h
```

→ Establecemos el usuario para el acceso web

```
### T-Pot User Configuration ...  
### Enter your web user name: tpotuser  
### Your username is: tpotuser  
y### Is this correct? (y/n) y  
  
### Enter password for your web user: [ ]
```

```
[+] Pulling 12/38
  ✓ tanner_redis Skipped - Image is already being pulled by map_redis
  ✓ map_data Skipped - Image is already being pulled by map_web
  ✓ conpot_kamstrup_382 Skipped - Image is already being pulled by conpot_guardian_ast
  ✓ conpot_IEC104 Skipped - Image is already being pulled by conpot_guardian_ast
  ✓ tanner_api Skipped - Image is already being pulled by tanner
  ✓ conpot_ipmi Skipped - Image is already being pulled by conpot_guardian_ast
  " sentrypeer [##] 6.953MB / 7.462MB Pulling
  " ipphoney [#####] 47.54MB / 48.52MB Pulling
  " ewsposter [ ] Pulling
  " dionaea [ ] Pulling
  " medpot [ ] Pulling
  " logstash [ ] Pulling
  " wordpot [ ] Pulling
  " elasticsearch [ ] Pulling
  " tptotinit [#####] 23.69MB / 28.08MB Pulling
  " dicompot [ ] Pulling
  " mailoney [ ] Pulling
  " honeytrap [ ] Pulling
  " ... [ ] Pulling
```

**puerto 64295**

```

### Please review for possible honeypot port conflicts.
### While SSH is taken care of, other services such as
### SMTP, HTTP, etc. might prevent T-Pot from starting.

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State       User        Inode      PID/Program name
tcp        0      0 0.0.0.0:64295          0.0.0.0:*               LISTEN      0           36811      7565/sshd: /usr/sbi
tcp6       0      0 :::64295               :::*                    LISTEN      0           36822      7565/sshd: /usr/sbi
udp        0      0 0.0.0.0:68            0.0.0.0:*               0           15173      373/dhclient

### Done. Please reboot and re-connect via SSH on tcp/64295.

```

→ Entramos a la carpeta de tpot y actualizamos la aplicación

```
keepcoding@tpot:~/tpotce$ ls
CHANGELOG.md  data          docker        env.example   install.sh    README.md     uninstall.sh
CITATION.cff  deploy.sh     docker-compose.yml  genuser.sh    installer     SECURITY.md    update.sh
compose       doc           dps.ps1       genuserwin.ps1 LICENSE        tools         version
keepcoding@tpot:~/tpotce$ sudo ./update.sh -y
This script should not be run as root. Please run it as a regular user.

keepcoding@tpot:~/tpotce$ ./update.sh -y

#####

### Checking for version tag ...
#### 24.04.0 is eligible for the update procedure. [ OK ]
```

→Una vez finalizado iniciamos el servicio tpot y los servicios de docker

```
### Restoring T-Pot config file .env

### Done. You can now start T-Pot using 'systemctl start tpot' or 'docker compose up -d'.

keepcoding@tpot:~/tpotce$
```

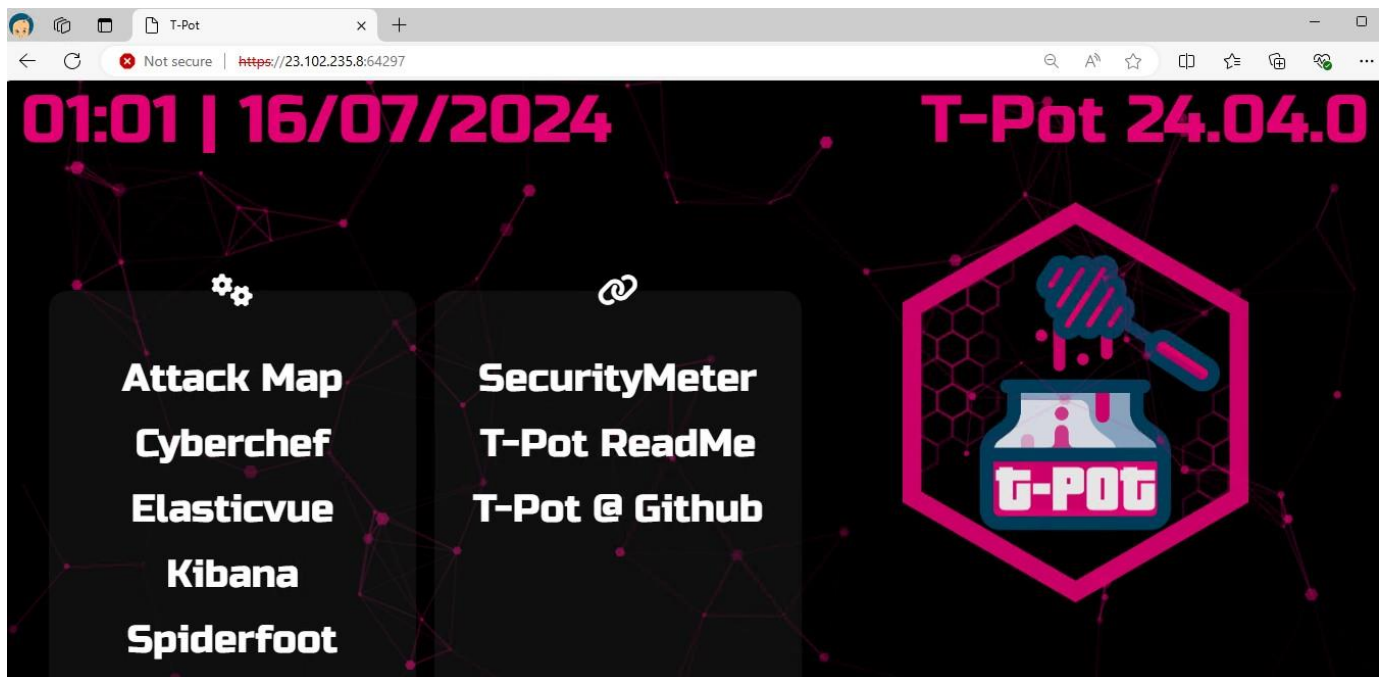
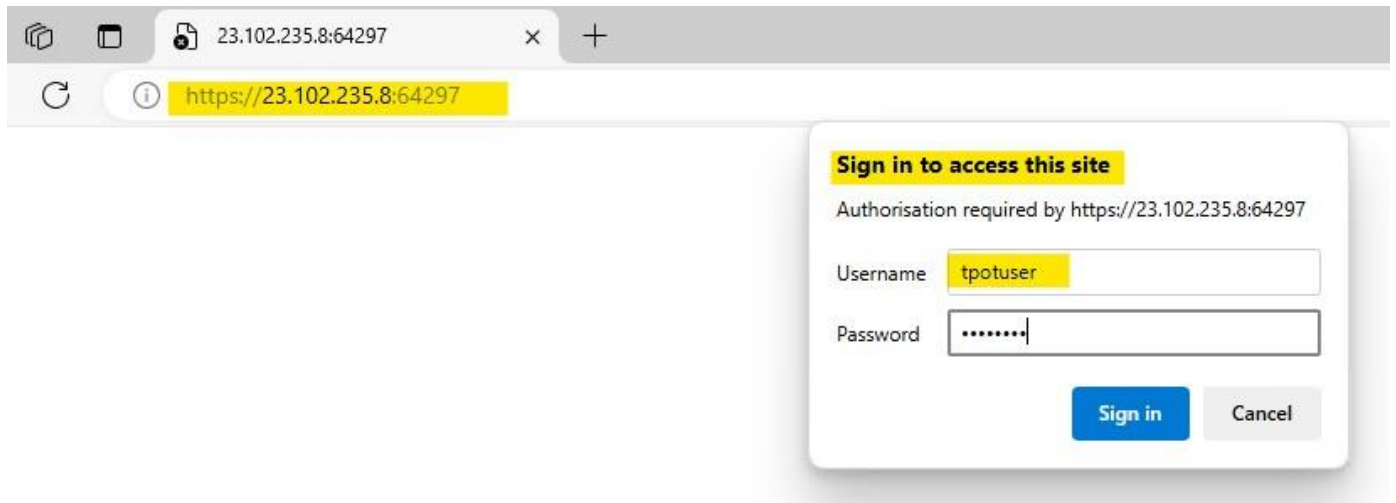
```
sudo systemctl start tpot
```



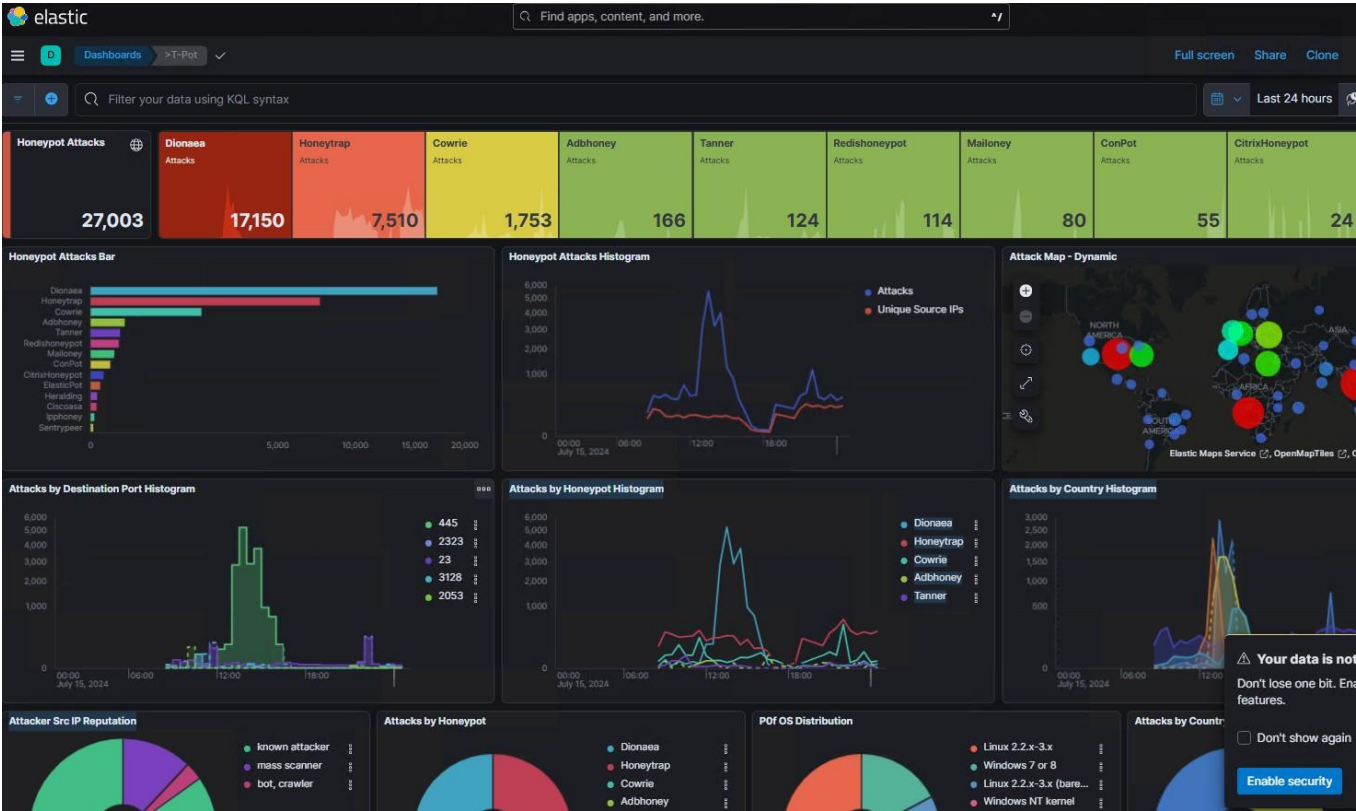
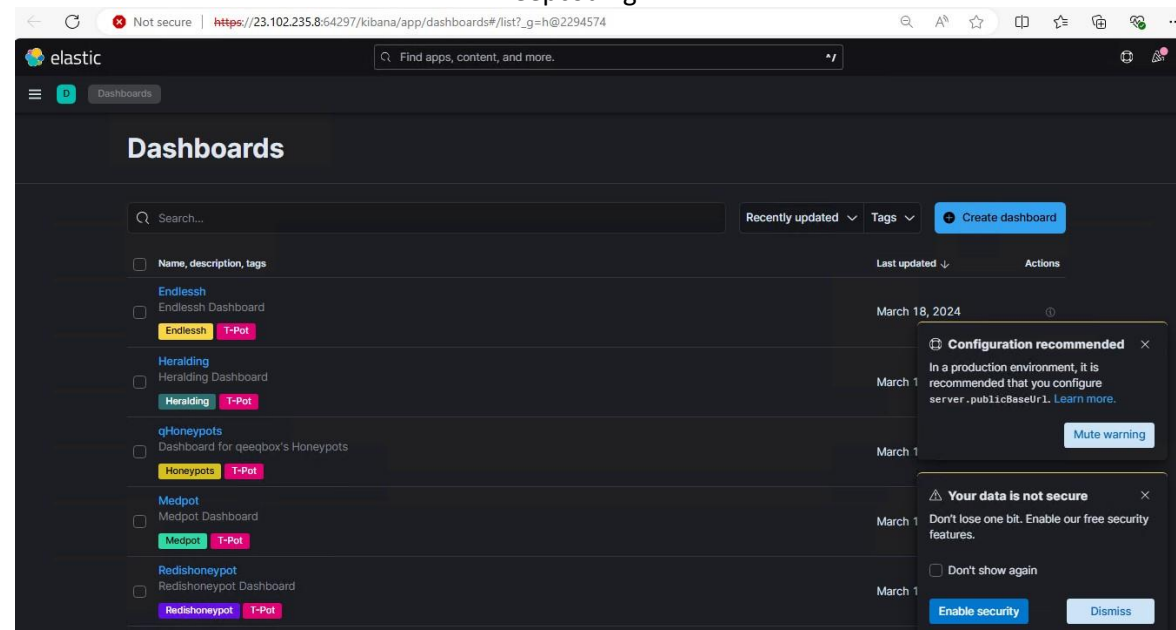
```
sudo docker compose up -d
```

Accedemos al portal a través del navegador con la IP pública de la máquina y el puerto **64297**

```
https://23.102.235.8:64297
```



Accedemos a Kibana



Carlos Gutierrez Torrejon  
Álvaro García De La  
Mata

keepcoding

