KENSEI
CYBERSECURITY

APRESENTAÇÃO TÉCNICA

ANÁLISE DE SEGMENTAÇÃO DE REDE AMBIENTE SIMULADO

Aluno: Fabio Sales
Professor: Jose Menezes
Instrutores: Gilson/ João

Apresentação Técnica – Análise de Segmentação de Rede em Ambiente Simulado

Autor: Fabio Salomão de Oliveira Sales

Analista em formação – Cibersegurança – Kensei – V.N.W.

Dados: 28 de julho de 2025
Versão: 1.0

Sumário Executivo

Esse relatório tem por finalidade trazer um apanhado geral dos principais achados da análise que fizemos em um ambiente simulado com Docker, onde testamos a segmentação entre três redes encontradas:

- corp_net (onde rola o funcionamento principal da empresa),
- guest_net (espaço feito pros visitantes se conectarem),
- infra_net (parte mais crítica, que cuida dos serviços internos e sistemas sensíveis da organização).

Durante o teste, usamos ferramentas como Nmap e Rustscan, junto com outras que ajudam a mapear melhor a rede. E com base nos resultados encontramos alguns pontos que acenderam o alerta de possíveis riscos.

Inicialmente, não tinha uma separação firme entre as redes. Isso é um grande problema, porque acaba permitindo que áreas que deveriam estar isoladas conversem entre si. Um exemplo grave: conseguimos sair da guest_net e acessar recursos importantes lá da corp_net — algo que, em um ambiente real, pode botar tudo a perder.

Outro ponto preocupante foram as várias portas abertas sem necessidade e clareza. Isso é praticamente um convite a qualquer atacante fazer varredura e explorar o que tiver pela frente. Além do mais, foi observado que não tinha nenhum tipo de monitoramento entre as redes, fato é que deixa tudo ainda mais vulnerável, já que fica difícil perceber quando alguém está tentando fazer algo errado.

Diante disso, é recomendado que algumas ações sejam tomadas imediatamente:

- Reforçar as regras de firewall, usando o princípio do menor privilégio (só acessa o que realmente precisa);
- Separar as redes com VLANs e firewalls internos, controlando melhor quem pode falar com quem;
- Isolar bem as redes fisicamente e logicamente, pra impedir movimentações laterais de possíveis invasores;
- Atualizar tudo que está desatualizado, desde sistemas até bibliotecas;
- Incluir ferramentas que fiquem de olho em vulnerabilidades o tempo todo, como parte da rotina da equipe de segurança.

Em resumo: hoje, qualquer pessoa que entrar na guest_net tem caminho aberto até partes críticas da infraestrutura. Isso mostra que a arquitetura da rede precisa ser revisada com urgência.

O ambiente como está apresenta vários riscos sérios, que podem ser explorados tanto por quem está de fora quanto por alguém mal-intencionado de dentro da rede. Aplicar as recomendações que estão aqui é essencial visando proteger melhor os dados da empresa e garantir uma postura mais segura no dia a dia.

Fábio Sales
Analista em formação – Cibersegurança

Objetivo do Relatório

Analisar a rede simulada para identificação de exposição, segmentação e riscos operacionais.

Com base nessa análise, o objetivo é apontar as falhas encontradas e sugerir melhorias práticas pra deixar o ambiente ficar mais seguro, evitar acessos indevidos e fortalecer a proteção dos dados e sistemas internos. Tudo isso de forma clara, direta e aplicável.

Este relatório tem como objetivo documentar, verificar ativos e analisar as interfaces de rede, os endereçamentos IP e os serviços expostos em um ambiente de rede simulado. A finalidade é compreender a estrutura de rede e das sub-redes, identificar potenciais vulnerabilidades associadas a cada interface como portas abertas e propor medidas de mitigação de possíveis danos, baseado nas boas práticas e também destacar possíveis vulnerabilidades poderiam ser exploradas, com foco em aprendizado prático de análise e estudo para formação do Modulo-1 de Cyber Segurança da Kensei - Vai Na Web.

Escopo

Ambiente docker simulado com múltiplos hosts e redes segmentadas.

O ambiente usa Docker para simular vários computadores conectados em redes separadas. O objetivo é criar um espaço para testar a comunicação entre eles, ver como as redes segmentadas funcionam e encontrar possíveis problemas, tudo de forma segura e isolada.

O escopo contempla:

- A definição da ferramenta e metodologia utilizada (RustScan);
- A identificação de hosts ativos e respectivas portas abertas;
- A inferência de possíveis serviços baseados nas portas detectadas;
- A recomendação de ações para mitigar riscos operacionais;
- A organização dos dados em formato de inventário técnico para consulta e posterior investigação.

Este trabalho faz parte do módulo de estudo prático em Análise de Rede e Segmentação em Cibersegurança, com enfoque em ambientes simulados que reproduzem cenários reais de redes corporativas.

INTERFACES ENCONTRADAS

Logo após o acesso a ‘’analyst’’ foi encontrado com o comando ‘’ ip a ‘’

Encontrados:

- **1 interface de loopback (lo)**
- **3 interfaces Ethernet (eth0, eth1, eth2)**, todas configuradas com IPs privados em diferentes sub-redes:
    - eth0 → 10.10.10.2/24
    - eth1 → 10.10.50.6/24
    - eth2 → 10.10.30.2/24

Essas interfaces podem ser utilizadas para administração, tráfego interno, ou isolamento de serviços.

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host proto kernel_lo
       valid_lft forever preferred_lft forever
2: eth0@if20: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 76:71:52:39:42:86 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.10.10.2/24 brd 10.10.10.255 scope global eth0
       valid_lft forever preferred_lft forever
3: eth1@if29: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether ce:87:1d:62:b9:da brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.10.50.6/24 brd 10.10.50.255 scope global eth1
       valid_lft forever preferred_lft forever
4: eth2@if32: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 32:aa:ed:7e:10:00 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.10.30.2/24 brd 10.10.30.255 scope global eth2
       valid_lft forever preferred_lft forever
```

Na imagem acima demonstra as interfaces de rede e suas configurações.

O local host em 127.0.0.1 que não se comunica com outras máquinas, somente com ela mesma.

eth0@if20 uma placa de rede conectada à rede 10.10.10.0/24, tem o IP 10.10.10.2 com máscara /24 -> rede 10.10.10.0 a 10.10.10.255.

eth1@if29, que está conectada à rede 10.10.50.0/24, com o IP: 10.10.50.6, Segundo o auxilio da IA a eth1 ‘’pode estar sendo usada para tráfego separado, por exemplo, DMZ, backups ou serviços internos.’’

eth2@if32 está conectada à rede 10.10.30.0/24, com o IP: 10.10.30.2, segundo o auxílio da IA, a eth2@if32 Pode estar conectada a uma terceira rede, possivelmente usada para administração, monitoramento ou isolamento de tráfego.

A interface lo, também chamada de *loopback*, é uma parte padrão que existe em qualquer sistema operacional moderno. Ela serve basicamente para o próprio sistema conversar consigo mesmo — como se fosse uma ponte interna. O famoso IP 127.0.0.1 é

usado aqui, e é útil para testar serviços locais ou garantir que certas aplicações estejam rodando corretamente. Essa interface não traz riscos de segurança externos, mas é essencial pro bom funcionamento de programas e serviços dentro da própria máquina.

A interface eth0 está com o IP 10.10.10.4 e faz parte da rede 10.10.10.0/24. É provavelmente a principal conexão com a rede local ou com a internet, e por isso merece mais atenção. Como costuma ser a interface "de frente", por onde passam os acessos externos, é fundamental ter controle rígido de portas abertas, firewall bem configurado e autenticação segura. Qualquer descuido aqui pode ser uma brecha para invasores.

A interface eth1, com o IP 10.10.50.6, está ligada à rede 10.10.50.0/24. Essa rede pode ser uma DMZ — uma zona "neutra", onde ficam serviços que precisam ser acessados por fora, como servidores web, DNS ou FTP. A ideia de isolar essa parte é justamente evitar que, caso um desses serviços seja invadido, o atacante não tenha acesso direto ao coração da rede interna. Se essa máquina estiver fazendo papel de firewall ou roteador, o cuidado com as regras de acesso entre as redes precisa ser ainda maior.

A interface eth2, com IP 10.10.30.2, está na rede 10.10.30.0/24. Essa rede pode estar sendo usada para administração ou controle interno — o tipo de rede que só o pessoal autorizado deveria acessar. Coisas como SSH, monitoramento (Zabbix, por exemplo) ou sistemas de backup podem passar por aqui. Justamente por ser mais escondida, pode acabar sendo esquecida na hora de aplicar regras de segurança. Mas isso seria um erro. É importante garantir autenticação forte, criptografia e regras de firewall bem restritas pra evitar que essa via se torne uma porta de entrada.

Metodologia

- **Ferramentas: nmap, rustscan, netdiscover, ping, etc.**

  Usamos essas ferramentas para entender como o ambiente funciona e identificar possíveis pontos de atenção.

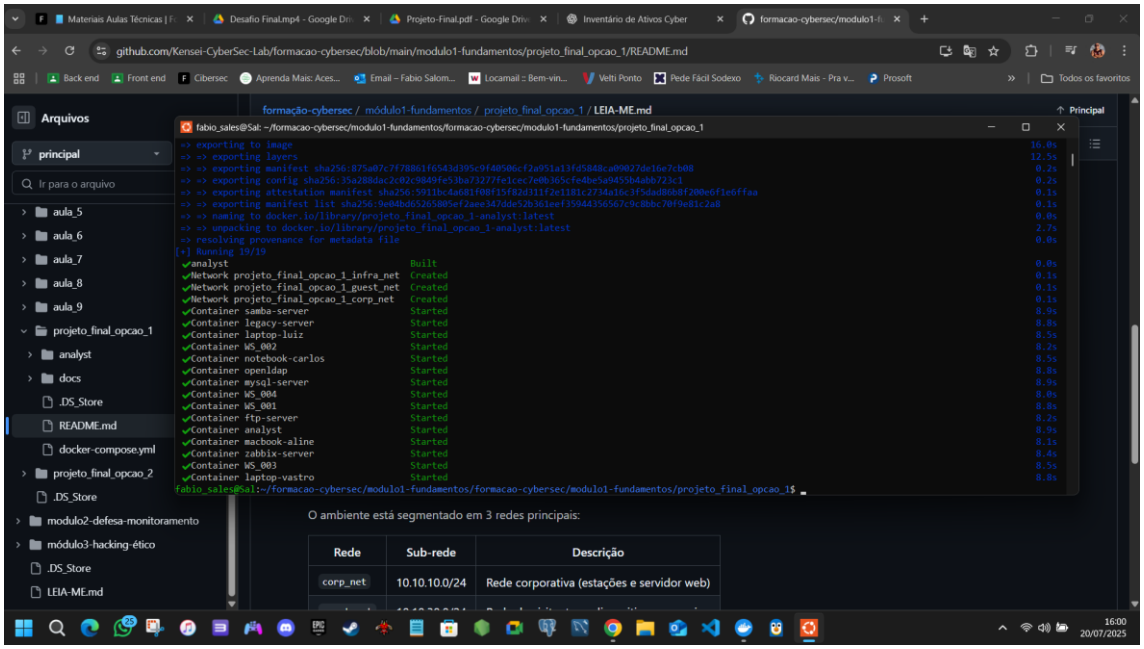
- **Coleta ativa de dados de rede**

  Fizemos uma coleta ativa de informações com as ferramentas acima citadas, testamos diretamente a rede para descobrir quais computadores estão conectados, quais serviços estão abertos e as portas utilizadas.

- **Análise manual e documentada**

  Depois, analisamos esses dados, registrando e documentando tudo de forma organizada criando um banco de ativos para entender melhor o que foi encontrado e poder dar recomendações para manter as redes mais seguras.

PEGAR INFORMAÇOES DAS REDES



Início APAGAR O COMPOSE - IMAGINE COMO SE FOSSE UMA REDE CORPORATIVA-----

Docker composse up -d

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ip a | grep inet
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host proto kernel_lo
    inet 10.10.10.2/24 brd 10.10.10.255 scope global eth0
    inet 10.10.50.6/24 brd 10.10.50.255 scope global eth1
    inet 10.10.30.2/24 brd 10.10.30.255 scope global eth2
```

Acima estão listadas todas as interfaces de rede e seus detalhes filtrados pelas linhas que contém endereços IP atribuídos as respectivas interfaces.

Relatório de Endereços IP - Reconhecimento de Redes

Data de geração: 20/07/2025 20:00:07

Interface	Endereço IP	Tipo
lo	127.0.0.1/8	IPv4
host	::1/128	IPv4
eth0	10.10.10.2/24	IPv4
eth1	10.10.50.6/24	IPv4
eth2	10.10.30.2/24	IPv4

Acima relatório informativo com ajuda de IA, com base nos dados do print com código ‘ip a.

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ip route
default via 10.10.10.1 dev eth0
10.10.10.0/24 dev eth0 proto kernel scope link src 10.10.10.2
10.10.30.0/24 dev eth2 proto kernel scope link src 10.10.30.2
10.10.50.0/24 dev eth1 proto kernel scope link src 10.10.50.6

┌──(root㉿85a984bf752c)-[/home/analyst]
└─#
```

Com base no auxílio da IA obtive a informação de **que as informações enviadas para uma rede externa como por exemplo (internet)** é enviada via gateway 10.10.10.1., e meu ip usado nesta rede é 10.10.10.2

A interface eth0 é a saída principal para fora da maquina, especificamente para a internet, as outras são interfaces de rede local

Segue tabela informativa gerada com auxilio da IA

Interface	IP Local	Rede/Sub-rede	Rota de Saída
lo	127.0.0.1	loopback	local apenas
eth0	10.10.10.2	10.10.10.0/24	**default gateway** (10.10.10.1)
eth1	10.10.50.6	10.10.50.0/24	local
eth2	10.10.30.2	10.10.30.0/24	local

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ip a | grep inet > recon-redes.txt

┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ls
recon-redes.txt

┌──(root㉿85a984bf752c)-[/home/analyst]
└─# cat recon-redes.txt
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host proto kernel_lo
    inet 10.10.10.2/24 brd 10.10.10.255 scope global eth0
    inet 10.10.50.6/24 brd 10.10.50.255 scope global eth1
    inet 10.10.30.2/24 brd 10.10.30.255 scope global eth2
```

Esse comando enumera as redes internas, coleta passivamente informações e cria um artefato de evidência (logs) em txt.

Após criarmos um arquivo para salvar as informações para um futuro reconhecimento /inventario/ base para escaneamento, usamos um ‘’cat’’ para verificar o conteúdo no arquivo.

TESTE DE CONECTIVIDADE DAS REDES

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ping -c 3 10.10.10.1  # corp_net
PING 10.10.10.1 (10.10.10.1) 56(84) bytes of data.
64 bytes from 10.10.10.1: icmp_seq=1 ttl=64 time=22.5 ms
64 bytes from 10.10.10.1: icmp_seq=2 ttl=64 time=0.164 ms
64 bytes from 10.10.10.1: icmp_seq=3 ttl=64 time=0.082 ms

--- 10.10.10.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2029ms
rtt min/avg/max/mdev = 0.082/7.574/22.477/10.537 ms

┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ping -c 3 10.10.30.1  # guest_net
PING 10.10.30.1 (10.10.30.1) 56(84) bytes of data.
64 bytes from 10.10.30.1: icmp_seq=1 ttl=64 time=11.5 ms
64 bytes from 10.10.30.1: icmp_seq=2 ttl=64 time=0.159 ms
64 bytes from 10.10.30.1: icmp_seq=3 ttl=64 time=0.126 ms

--- 10.10.30.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 0.126/3.944/11.548/5.376 ms

┌──(root㉿85a984bf752c)-[/home/analyst]
└─# ping -c 3 10.10.50.1  # infra_net
PING 10.10.50.1 (10.10.50.1) 56(84) bytes of data.
64 bytes from 10.10.50.1: icmp_seq=1 ttl=64 time=0.732 ms
64 bytes from 10.10.50.1: icmp_seq=2 ttl=64 time=0.144 ms
64 bytes from 10.10.50.1: icmp_seq=3 ttl=64 time=0.091 ms

--- 10.10.50.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2027ms
rtt min/avg/max/mdev = 0.091/0.322/0.732/0.290 ms
```

Foi verificado que através do Ping que os hosts estão ativos nas redes, isso nos ajudar a identificar as máquinas que estão acessíveis e que podem ser potenciais alvos para ataques. Caso o Ping falhe como não foi o caso poderíamos identificar também possíveis filtros ou firewalls bloqueando pacotes ICMP.

Obs.: O I**CMP**, ou Protocolo de Mensagens de Controle da Internet, é um protocolo de rede usado para diagnosticar problemas de comunicação e relatar erros em redes IP. (informativo retirado em busca na plataforma google.com)

DESCOBRIR OS HOSTS COM NMAP, PING e SCAN

Redes: corp_net, guest_net e infra_net

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# nmap -sn -T4 10.10.10.0/24 -oG - | grep "Up"
Host: 10.10.10.1 ()     Status: Up
Host: 10.10.10.10 (WS_001.projeto_final_opcao_1_corp_net)       Status: Up
Host: 10.10.10.101 (WS_002.projeto_final_opcao_1_corp_net)      Status: Up
Host: 10.10.10.127 (WS_003.projeto_final_opcao_1_corp_net)      Status: Up
Host: 10.10.10.222 (WS_004.projeto_final_opcao_1_corp_net)      Status: Up
Host: 10.10.10.2 (85a984bf752c) Status: Up
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.30.0/24 -oG - | grep "Up"
Host: 10.10.30.1 ()     Status: Up
Host: 10.10.30.10 (ftp-server.projeto_final_opcao_1_infra_net)  Status: Up
Host: 10.10.30.11 (mysql-server.projeto_final_opcao_1_infra_net)        Status: Up
Host: 10.10.30.15 (samba-server.projeto_final_opcao_1_infra_net)        Status: Up
Host: 10.10.30.17 (openldap.projeto_final_opcao_1_infra_net)    Status: Up
Host: 10.10.30.117 (zabbix-server.projeto_final_opcao_1_infra_net)      Status: Up
Host: 10.10.30.227 (legacy-server.projeto_final_opcao_1_infra_net)      Status: Up
Host: 10.10.30.2 (1c91b12c1330) Status: Up
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.50.0/24 -oG - | grep "Up"
Host: 10.10.50.1 ()     Status: Up
Host: 10.10.50.2 (macbook-aline.projeto_final_opcao_1_guest_net)        Status: Up
Host: 10.10.50.3 (notebook-carlos.projeto_final_opcao_1_guest_net)      Status: Up
Host: 10.10.50.4 (laptop-luiz.projeto_final_opcao_1_guest_net)  Status: Up
Host: 10.10.50.5 (laptop-vastro.projeto_final_opcao_1_guest_net)        Status: Up
Host: 10.10.50.6 (1c91b12c1330) Status: Up
```

Acima verificamos os hosts ativos nas redes '' 10.10.10.0/24, 10.10.30.0/24 e 10.10.50.0/24'', quais IP's dentro desses intervalos estão respondendo justamente **conectados às redes** no momento.

GERANDO INVENTARIO DE IPS ATIVOS NAS REDES

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# nmap -sn -T4 10.10.10.0/24 -oG - | awk '/Up$/{print $2}' | tee corp_net_ips.txt
10.10.10.1
10.10.10.10
10.10.10.101
10.10.10.127
10.10.10.222
10.10.10.2
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.30.0/24 -oG - | awk '/Up$/{print $2}' | tee infra_net_ips.txt
10.10.30.1
10.10.30.10
10.10.30.11
10.10.30.15
10.10.30.17
10.10.30.117
10.10.30.227
10.10.30.2
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.50.0/24 -oG - | awk '/Up$/{print $2}' | tee guest_net_ips.txt
10.10.50.1
10.10.50.2
10.10.50.3
10.10.50.4
10.10.50.5
10.10.50.6
```

Nas etapas acima geramos uma lista limpa apenas com os endereços IP desses hosts. A lista é exibida no terminal e no mesmo tempo cria e salva em um arquivo chamado corp_net_ips.txt, infra_net_ips.txt e guest_net_ips.txt para cada rede, mantendo também a exibição no terminal com tee.

GERANDO INVENTARIO DE IPS ATIVOS COM NOMES DOS HOSTS

```
┌──(root㉿85a984bf752c)-[/home/analyst]
└─# nmap -sn -T4 10.10.10.0/24 -oG - | awk '/Up$/{print $2, $3}' | tee corp_net_ips_hosts.txt
10.10.10.1 ()
10.10.10.10 (WS_001.projeto_final_opcao_1_corp_net)
10.10.10.101 (WS_002.projeto_final_opcao_1_corp_net)
10.10.10.127 (WS_003.projeto_final_opcao_1_corp_net)
10.10.10.222 (WS_004.projeto_final_opcao_1_corp_net)
10.10.10.2 (85a984bf752c)

┌──(root㉿85a984bf752c)-[/home/analyst]
└─#
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.30.0/24 -oG - | awk '/Up$/{print $2, $3}' | tee infra_net_ips_hosts.txt
10.10.30.1 ()
10.10.30.10 (ftp-server.projeto_final_opcao_1_infra_net)
10.10.30.11 (mysql-server.projeto_final_opcao_1_infra_net)
10.10.30.15 (samba-server.projeto_final_opcao_1_infra_net)
10.10.30.17 (openldap.projeto_final_opcao_1_infra_net)
10.10.30.117 (zabbix-server.projeto_final_opcao_1_infra_net)
10.10.30.227 (legacy-server.projeto_final_opcao_1_infra_net)
10.10.30.2 (1c91b12c1330)
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -sn -T4 10.10.50.0/24 -oG - | awk '/Up$/{print $2, $3}' | tee guest_net_ips_hosts.txt
10.10.50.1 ()
10.10.50.2 (macbook-aline.projeto_final_opcao_1_guest_net)
10.10.50.3 (notebook-carlos.projeto_final_opcao_1_guest_net)
10.10.50.4 (laptop-luiz.projeto_final_opcao_1_guest_net)
10.10.50.5 (laptop-vastro.projeto_final_opcao_1_guest_net)
10.10.50.6 (1c91b12c1330)
```

Acima podemos verificar uma varredura de dispositivos ativos (hosts "Up") nas redes verificadas nesse relatório onde o comando nos gera uma lista contendo o endereço IP e o nome do host (quando disponível), salvando essa informação em um arquivo chamado corp..., infra... e guest_net_ips_hosts.txt em cada teste.

Comandos:

nmap -sn -T4 10.10.10.0/24 -oG - | grep "Up"

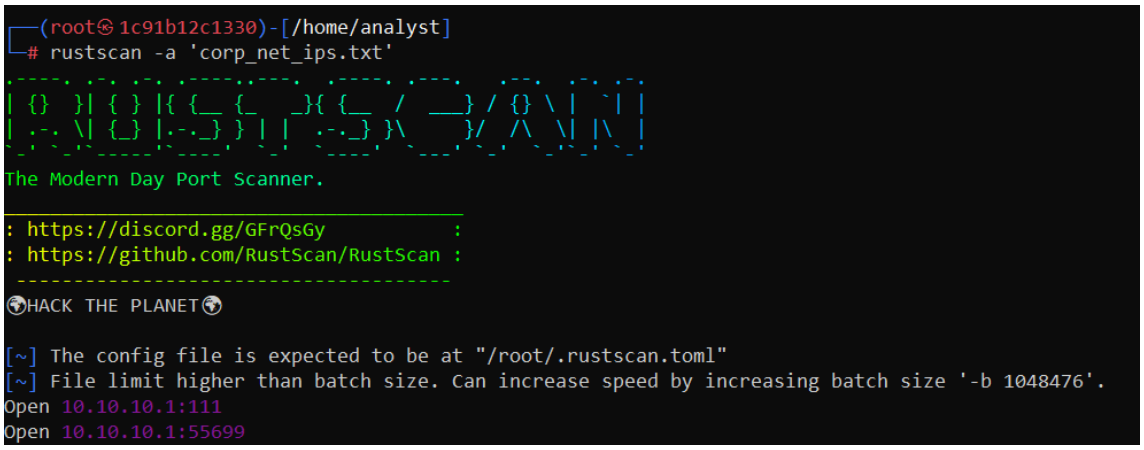
nmap -sn -T4 10.10.10.0/24 -oG - | awk '/Up$/{print $2}' | tee corp_net_ips.txt

nmap -sn -T4 10.10.10.0/24 -oG - | awk '/Up$/{print $2, $3}' | tee corp_net_ips_hosts.txt

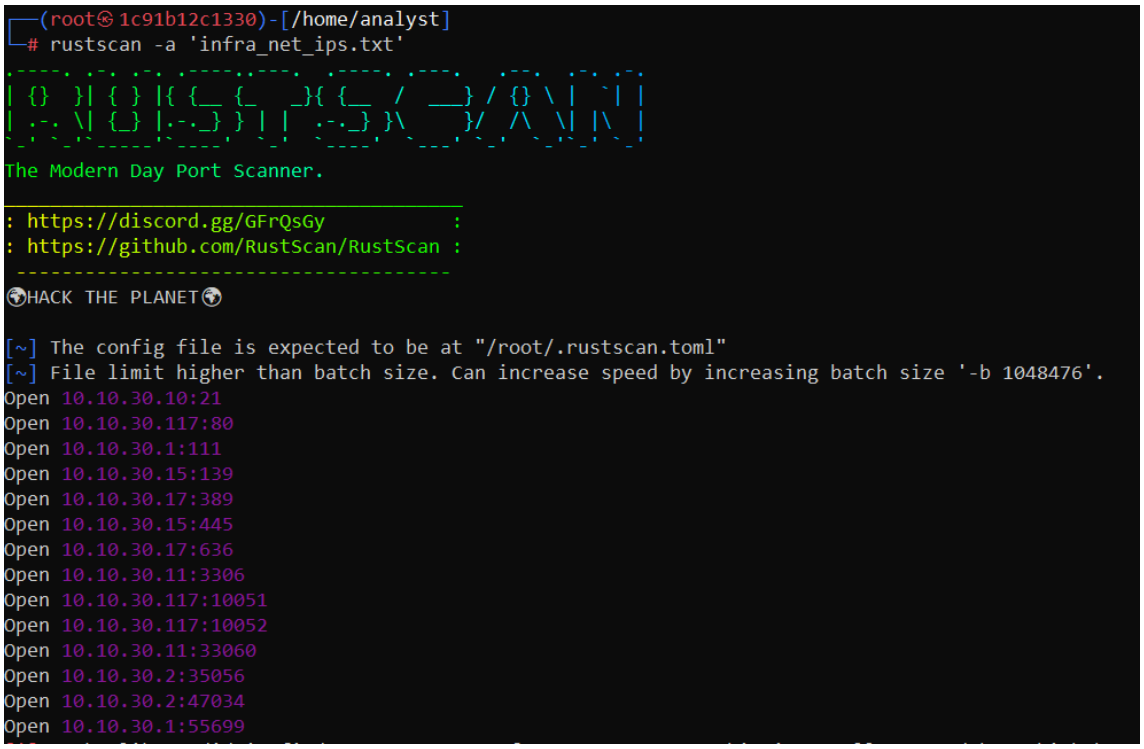
Esses três comandos são utilizados para reconhecimento e coleta de informações básicas sobre uma rede, etapa fundamental tanto para o Red Team (ataque) quanto para o Blue Team (defesa).

Para o Red Team podemos verificar quem está online na rede e já no Blue Team se tem hosts indevidos conectados na mesma rede, tudo gerando arquivos de inventario dos ips ativos com ou sem os nomes nos hosts para relatórios e comprovação documental das verificações caso precise de uma auditoria.

SCAN RÁPIDO COM RUSTSCAN PARA PEGAR AS PORTAS ABERTAS



No escaneamento rápido da rede corp_net, não foram identificadas máquinas com portas abertas que possivelmente venham a ser vulnerabilidades, encontramos somente o roteador da rede no endereço 10.10.10.1



No escaneamento realizado com RustScan na rede infra_net (**10.10.30.0/24)**, foram identificadas diversas máquinas com portas abertas que podem representar potenciais pontos de atenção para segurança.

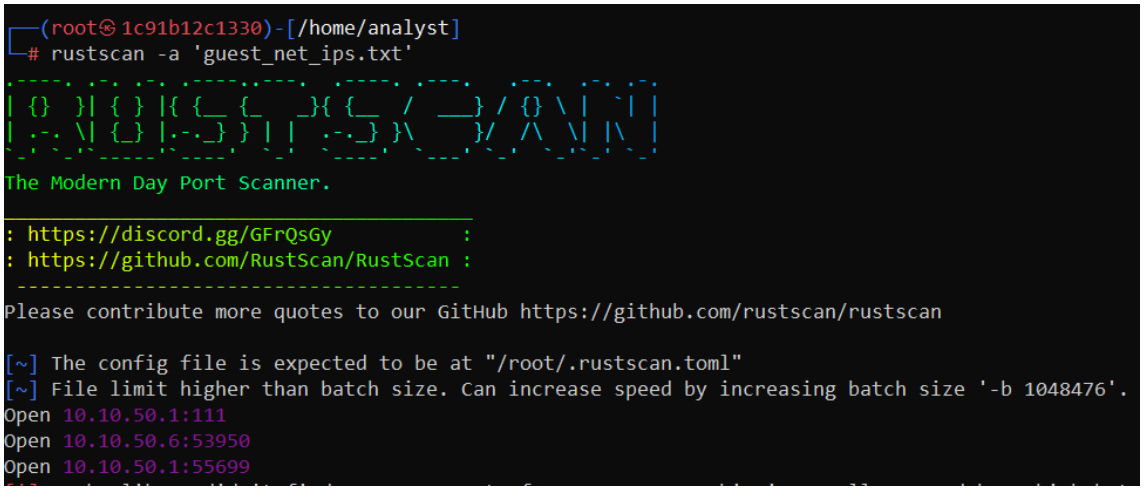
Os hosts ativos que responderam na varredura apresentaram as seguintes portas abertas na rede:

- **10.10.30.10**: porta TCP **21** (possivelmente serviço FTP)
- **10.10.30.117**: porta TCP **80** (HTTP)
- **10.10.30.1**: porta TCP **111** (RPCbind ou portmap)
- **10.10.30.15**: portas TCP **139**, **445** (provavelmente SMB, compartilhamento de arquivos).
- **10.10.30.17**: portas TCP **389**, **636** (LDAP e LDAPS)
- **10.10.30.11**: portas TCP **3306**, **33060** (MySQL e MySQL X Protocol)
- **10.10.30.117**: portas TCP **10051**, **10052** (serviços relacionados ao Zabbix Agent)
- **10.10.30.2**: portas TCP **35056**, **47084**
- **10.10.30.1**: porta TCP **55699** (porta não comum, deve ser investigada)

A imagem mostrou na saída da ferramenta RustScan com os IPs e as respectivas portas abertas identificadas.

Essa varredura indica que a rede infra_net apresenta **diversos serviços expostos**, muitos deles sem validação de segurança imediata. Destacam-se especialmente os serviços de FTP, SMB, LDAP e MySQL, que são frequentemente explorados em ataques se mal configurados ou desatualizados.

É recomendado realizar uma análise mais profunda dos serviços em execução, verificar se estão devidamente atualizados e protegidos com autenticação forte, além de revisar as regras de firewall e segmentação para minimizar riscos operacionais.



No escaneamento rápido da rede guest_net, não foram identificadas máquinas com portas abertas que possivelmente venham a ser vulnerabilidades, encontramos somente o roteador da rede no endereço 10.10.50.1 e, a análise indicou também que o host 10.10.50.6 está ativo e respondeu à requisição, e o mesmo conclui-se que a porta TCP 53950 foi identificada como fechada conforme imagem abaixo.

```
[~] Starting Script(s)
[>] Script to be run Some("nmap -vvv -p {{port}} {{ip}}")

[~] Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-24 01:57 UTC
Initiating SYN Stealth Scan at 01:57
Scanning 1c91b12c1330 (10.10.50.6) [1 port]
Completed SYN Stealth Scan at 01:57, 0.02s elapsed (1 total ports)
Nmap scan report for 1c91b12c1330 (10.10.50.6)
Host is up, received localhost-response (0.000069s latency).
Scanned at 2025-07-24 01:57:46 UTC for 0s
```

```
PORT      STATE  SERVICE REASON
53950/tcp closed unknown reset ttl 64
```

```
Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
           Raw packets sent: 1 (44B) | Rcvd: 2 (84B)


┌──(root㉿1c91b12c1330)-[/home/analyst]
└─#
```

Verificação de acesso anônimo (porta 21 - host 10.10.30.10)

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -p 21 --script ftp-anon 10.10.30.10
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-24 02:14 UTC
Nmap scan report for ftp-server.projeto_final_opcao_1_infra_net (10.10.30.10)
Host is up (0.00011s latency).

PORT   STATE SERVICE
21/tcp open  ftp
MAC Address: 76:0E:51:74:42:9E (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.23 seconds
```

Com base no escana acima detectamos que o servidor FTP não está permitindo **acesso anônimo**, onde poderia ser uma **falha grave de segurança**, pois qualquer pessoa na rede poderia: Listar arquivos, fazer upload/download de dados sensíveis ou ate explorar o servidor como ponto de entrada na rede.

Verificação de serviço e versão rodando na porta 3306.

No escaneamento acima verificamos com o auxílio da IA, que o host está ativo, porta 3306 aberta, serviço MySQL acessível, a versão detectada foi MySQL 8.0.42, thread ID e capabilities listados nos mostra como o servidor está configurado internamente e autenticação: Salt: valor usado para criptografar a senha e Auth Plugin: caching_sha2_password — padrão em versões mais novas do MySQL. Lembrando que atualmente o MySQL está na versão 8.4.5 portanto necessitando de que seja realizado um update para segurança do sistema.

Verificação do servidor LDAP (IP: 10.10.30.17) porta 389

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -p 389 --script ldap-rootdse 10.10.30.17
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-24 02:16 UTC
Nmap scan report for openldap.projeto_final_opcao_1_infra_net (10.10.30.17)
Host is up (0.00011s latency).

PORT    STATE SERVICE
389/tcp open  ldap
| ldap-rootdse:
| LDAP Results
|   <ROOT>
|       namingContexts: dc=example,dc=org
|       supportedControl: 2.16.840.1.113730.3.4.18
|       supportedControl: 2.16.840.1.113730.3.4.2
|       supportedControl: 1.3.6.1.4.1.4203.1.10.1
|       supportedControl: 1.3.6.1.1.22
|       supportedControl: 1.2.840.113556.1.4.319
|       supportedControl: 1.2.826.0.1.3344810.2.3
|       supportedControl: 1.3.6.1.1.13.2
|       supportedControl: 1.3.6.1.1.13.1
|       supportedControl: 1.3.6.1.1.12
|       supportedExtension: 1.3.6.1.4.1.1466.20037
|       supportedExtension: 1.3.6.1.4.1.4203.1.11.1
|       supportedExtension: 1.3.6.1.4.1.4203.1.11.3
|       supportedExtension: 1.3.6.1.1.8
|       supportedLDAPVersion: 3
|       supportedSASLMechanisms: SCRAM-SHA-1
|       supportedSASLMechanisms: SCRAM-SHA-256
|       supportedSASLMechanisms: GS2-IAKERB
|       supportedSASLMechanisms: GS2-KRB5
|       supportedSASLMechanisms: GSS-SPNEGO
|       supportedSASLMechanisms: GSSAPI
|       supportedSASLMechanisms: DIGEST-MD5
|       supportedSASLMechanisms: OTP
|       supportedSASLMechanisms: CRAM-MD5
|       supportedSASLMechanisms: NTLM
|_      subschemaSubentry: cn=Subschema
MAC Address: FA:81:F0:4E:4B:17 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

Abaixo seguem os riscos identificados no servidor LDAP (IP: 10.10.30.17) conforme imagem acima, obtida no resultado do escaneamento com Nmap na porta 389 com ajuda interpretativa da IA:

Serviço LDAP exposto na rede: A porta 389 (LDAP) está aberta e acessível para qualquer um na rede. Permitindo que atacantes maliciosos tentem se conectar ao serviço e descobrir mais sobre o ambiente interno.

Acesso sem login (bind anônimo): O servidor está permitindo acesso sem precisar de usuário e senha. Isso facilita a coleta de informações sensíveis, como estrutura de diretório, nomes de domínio e até usuários, sem precisar invadir nada.

Métodos de login antigos e inseguros: Foram encontrados métodos de autenticação considerados fracos, como NTLM e CRAM-MD5. Eles podem ser explorados por atacantes para capturar ou quebrar senhas com ataques de força bruta ou de repetição.

Informações demais para quem não deveria ver, mesmo sem estar logado, o servidor revela muitos detalhes técnicos sobre sua configuração. Um atacante pode usar isso para planejar melhor uma invasão, domínio genérico mostra possível má configuração, o domínio dc=example,dc=org é padrão e geralmente usados em testes. Isso pode indicar que o servidor foi mal configurado, esquecido ou deixado vulnerável sem intenção.

Resumo:

Esse servidor LDAP está aberto demais e sem controle. Ele permite acesso sem senha, aceita formas antigas de autenticação e revela dados que deveriam estar protegidos. Isso tudo aumenta muito o risco de ataque e vazamento de informações internas.

Scan da porta 445, usada pelo protocolo SMB.

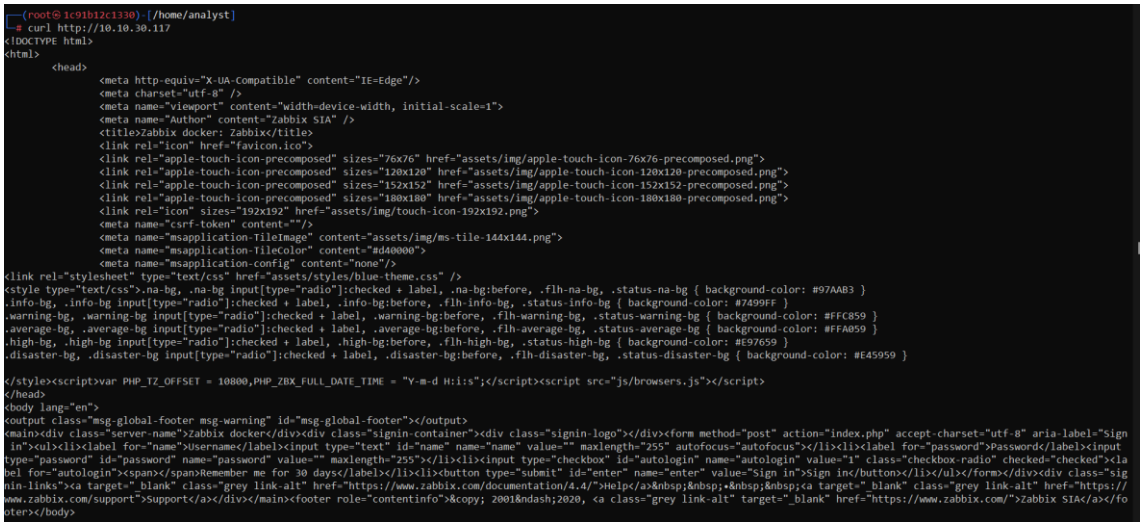
```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# nmap -p 445 --script smb-os-discovery,smb-enum-shares 10.10.30.15
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-24 02:17 UTC
Nmap scan report for samba-server.projeto_final_opcao_1_infra_net (10.10.30.15)
Host is up (0.000083s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 72:CC:7C:4C:ED:E0 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

O scan acima tentou descobrir o sistema operacional e as pastas compartilhadas do servidor no IP 10.10.30.15, mas o servidor não revelou nada útil — o que pode ser bom sinal de segurança.

No entanto, o simples fato de o serviço SMB estar exposto já exige atenção, pois se ele tiver falhas, pode ser usado como porta de entrada para ataques sérios, como roubo de arquivos, captura de senhas ou movimentação lateral dentro da rede.

Verificação de conteúdo completo das páginas/arquivos dom (curl / curl -i)

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# curl -I http://10.10.30.117
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 24 Jul 2025 02:18:16 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
Keep-Alive: timeout=20
X-Powered-By: PHP/7.3.14
Set-Cookie: PHPSESSID=82ae622f6a023a58546e3fc5006f656f; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN
```



Neste procedimento ‘curl’ nos dá várias informações sobre como o site funciona, qual tecnologia ele usa e, principalmente, quais medidas de segurança estão ativas para proteger tanto o servidor quanto o usuário com e sem cabeçalho de resposta com ‘-i’. Com isso por exemplo evita que se possa injetar código de JavaScrip malicioso, armazenar cookies na cache, miticador de ataques XSS e outros.

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# curl -I http://10.10.30.117 > infra_net_servico_webserver.txt
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
```

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# curl http://10.10.30.117 > infra_net_servico_zabbix.txt
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  3412    0  3412    0     0  63271      0 --:--:-- --:--:-- --:--:-- 64377
```

Nestes scans acima ele faz o mesmo procedimento do curl com e sem cabeçalho de resposta conforme mostrado na imagem anterior porem ela cria e guarda as informações num arquivo txt.

Verificando endereços físicos dos drivers de rede com o ARP "Address Resolution Protocol"

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# arp -a
macbook-aline.projeto_final_opcao_1_guest_net (10.10.50.4) at 32:6e:08:89:a9:7e [ether] on eth2
mysql-server.projeto_final_opcao_1_infra_net (10.10.30.11) at 02:28:7d:cc:72:19 [ether] on eth0
laptop-luiz.projeto_final_opcao_1_guest_net (10.10.50.2) at 76:16:e4:b4:fe:75 [ether] on eth2
samba-server.projeto_final_opcao_1_infra_net (10.10.30.15) at 72:cc:7c:4c:ed:e0 [ether] on eth0
legacy-server.projeto_final_opcao_1_infra_net (10.10.30.227) at 9e:a3:b2:b0:c4:c7 [ether] on eth0
zabbix-server.projeto_final_opcao_1_infra_net (10.10.30.117) at ae:1c:e2:2a:ad:d6 [ether] on eth0
openldap.projeto_final_opcao_1_infra_net (10.10.30.17) at fa:81:f0:4e:4b:17 [ether] on eth0
ftp-server.projeto_final_opcao_1_infra_net (10.10.30.10) at 76:0e:51:74:42:9e [ether] on eth0
notebook-carlos.projeto_final_opcao_1_guest_net (10.10.50.5) at ca:43:ba:a1:49:66 [ether] on eth2
laptop-vastro.projeto_final_opcao_1_guest_net (10.10.50.3) at ba:08:cf:2c:ce:ba [ether] on eth2
? (10.10.50.1) at a6:86:5f:17:2e:e9 [ether] on eth2
? (10.10.30.1) at 46:aa:13:29:06:14 [ether] on eth0
```

```
┌──(root㉿b09ab065d303)-[/home/analyst]
└─# nmap -sn 10.10.10.10
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 03:39 UTC
Nmap scan report for WS_001.projeto_final_opcao_1_corp_net (10.10.10.10)
Host is up (0.00013s latency).
MAC Address: 9A:72:72:C8:58:AF (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 0.14 seconds

┌──(root㉿b09ab065d303)-[/home/analyst]
└─# nmap -sn 10.10.10.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 03:40 UTC
Nmap scan report for WS_002.projeto_final_opcao_1_corp_net (10.10.10.101)
Host is up (0.000089s latency).
MAC Address: EA:67:34:5F:FE:ED (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds

┌──(root㉿b09ab065d303)-[/home/analyst]
└─# nmap -sn 10.10.10.127
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 03:41 UTC
Nmap scan report for WS_003.projeto_final_opcao_1_corp_net (10.10.10.127)
Host is up (0.00010s latency).
MAC Address: 46:35:5E:69:7D:35 (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds

┌──(root㉿b09ab065d303)-[/home/analyst]
└─# nmap -sn 10.10.10.222
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 03:42 UTC
Nmap scan report for WS_004.projeto_final_opcao_1_corp_net (10.10.10.222)
Host is up (0.00012s latency).
MAC Address: 42:CD:91:02:9A:E4 (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

Foi identificado nas sub redes os nomes dos dispositivos: macbook-aline, mysql-server, laptop-luiz, samba-server, legacy-server, zabbix-server, opendap, ftp-server, notebook-carlos e laptop-vastro, entre outros, fizemos um apanhado completo com todos os dados dos hosts onde compõem a rede geral e as sub redes.

```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# arp -a > recon_ip_maps.txt

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# cat /etc/resolv.conf
# Generated by Docker Engine.
# This file can be edited; Docker Engine will not make further changes once it
# has been modified.

nameserver 127.0.0.11
options ndots:0

# Based on host file: '/etc/resolv.conf' (internal resolver)
# ExtServers: [host(192.168.65.7)]
# Overrides: []
# Option ndots from: internal
```

Acima criamos um arquivo com o backup das informações em txt.

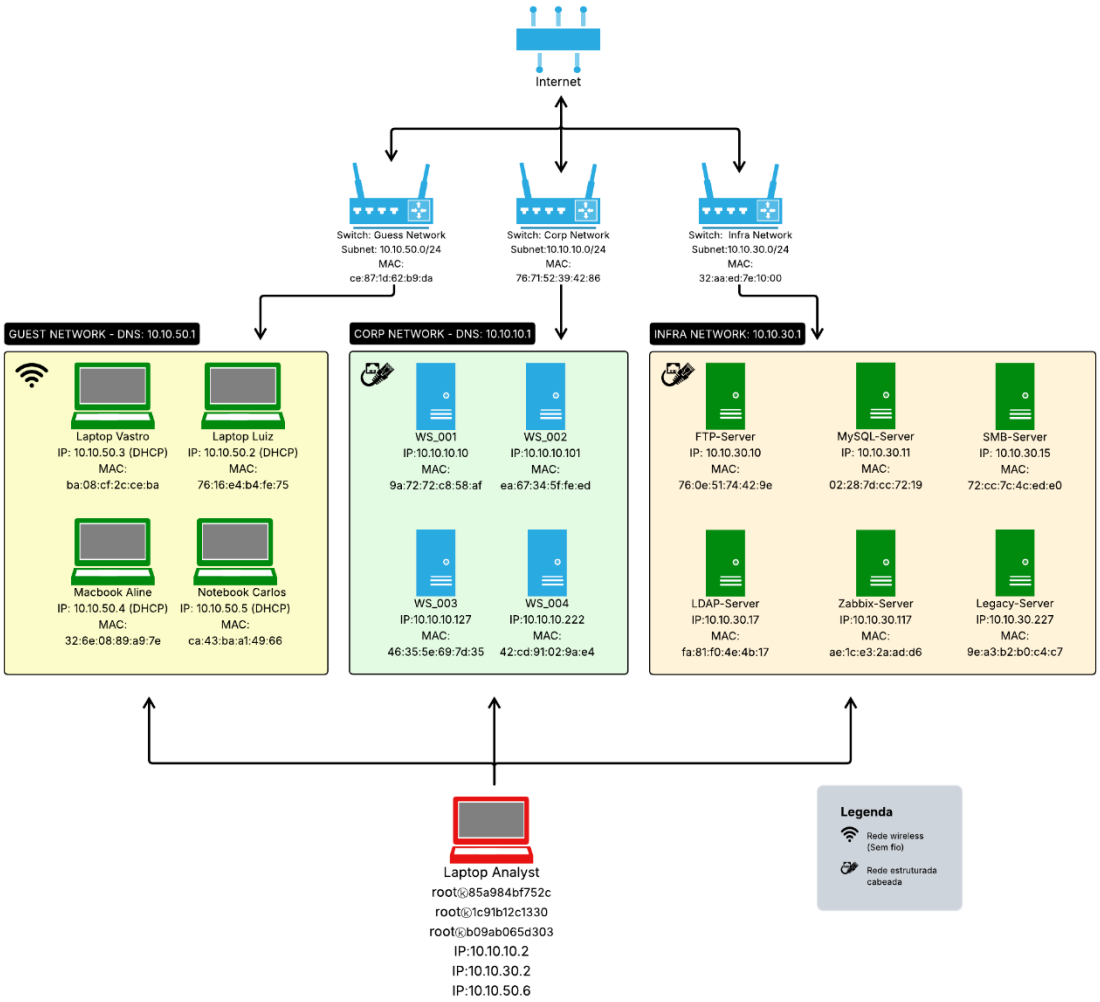
```
┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# mkdir -p /home/analyst/recon/{corp_net,guest_net,infra_net}

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# mv *corp*.txt /home/analyst/recon/corp_net/

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# mv *guest*.txt /home/analyst/recon/guest_net/

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# mv *infra*.txt /home/analyst/recon/infra_net/

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# mv *recon*.txt /home/analyst/recon/

┌──(root㉿1c91b12c1330)-[/home/analyst]
└─# exit
```

```
fabio_sales@Sal:~/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1$ docker cp analyst:/home/analyst/recon ./recon-backup
Successfully copied 24.6kB to /home/fabio_sales/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1/recon-backup
fabio_sales@Sal:~/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1$ ls -la
total 36
drwxr-xr-x  5 fabio_sales fabio_sales 4096 Jul 23 23:23 .
drwxr-xr-x 14 fabio_sales fabio_sales 4096 Jul 20 15:44 ..
-rw-r--r--  1 fabio_sales fabio_sales 6148 Jul 20 15:44 .DS_Store
-rw-r--r--  1 fabio_sales fabio_sales 2798 Jul 20 15:44 README.md
drwxr-xr-x  2 fabio_sales fabio_sales 4096 Jul 20 15:44 analyst
-rw-r--r--  1 fabio_sales fabio_sales    0 Jul 21 17:38 corp_net_ips_ports.txt
-rw-r--r--  1 fabio_sales fabio_sales 3137 Jul 20 15:44 docker-compose.yml
drwxr-xr-x  2 fabio_sales fabio_sales 4096 Jul 20 15:44 docs
drwxr-xr-x  5 fabio_sales fabio_sales 4096 Jul 23 23:22 recon-backup
```

```
fabio_sales@Sal:~/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1$ cd recon-backup
fabio_sales@Sal:~/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1/recon-backup$ ls -la
total 24
drwxr-xr-x 5 fabio_sales fabio_sales 4096 Jul 23 23:22 .
drwxr-xr-x 5 fabio_sales fabio_sales 4096 Jul 23 23:23 ..
drwxr-xr-x 2 fabio_sales fabio_sales 4096 Jul 23 23:22 corp_net
drwxr-xr-x 2 fabio_sales fabio_sales 4096 Jul 23 23:22 guest_net
drwxr-xr-x 2 fabio_sales fabio_sales 4096 Jul 23 23:22 infra_net
-rw-r--r-- 1 fabio_sales fabio_sales 1062 Jul 23 23:21 recon_ip_maps.txt
fabio_sales@Sal:~/formacao-cybersec/modulo1-fundamentos/formacao-cybersec/modulo1-fundamentos/projeto_final_opcao_1/recon-backup$
```

No passo acima estamos organizando os arquivos para um backup e para envio ou utilização em reuniões contendo todas as informações verificadas no escaneamento da rede.

Diagrama de Rede



Diagnóstico

A análise detalhada permitiu identificar dispositivos essenciais, serviços expostos e possíveis vulnerabilidades, servindo de base para recomendações práticas e um plano de ação direcionado.

Recomendações

As ações sugeridas envolvem:

- Reforço da segurança.
- Monitoramento contínuo da infraestrutura.
- Padronização e registro de processos.
- Execução de auditorias regulares.
- Adoção de medidas complementares para fortalecer a resiliência e a eficiência operacional.

Segurança & Monitoramento

Inclui ajustes na configuração de serviços expostos, controle de acessos, proteção contra ameaças e adoção de soluções de prevenção e detecção de intrusão (IPS/IDS). Ferramentas como Zabbix, Nagios e Prometheus foram indicadas para garantir o monitoramento em tempo real, com alertas automatizados para eventos críticos.

Documentação

Foram organizados procedimentos operacionais, inventário atualizado e registros de mudanças, promovendo clareza, controle e apoio em situações críticas.

Anexos

- Saída dos scans
- Impressões de ferramentas – (constam no corpo do relatório)
- Diagrama da rede – (consta no corpo do relatório)

Pontos de atenção para um possível 80/20

eth0	10.10.10.0/24	Principal,	Verificar regras de firewall e saída
eth1	10.10.50.0/24	Rede interna isolada,	Segmentar, auditar e proteger acesso
eth2	10.10.30.0/24	Gerência/serviços,	Monitoramento e controle de tráfego

Plano de Ação 80/20

Foco nas ações com maior impacto e menor esforço (Princípio de Pareto):		
Item	**Ação Imediata (80/20)**	**Justificativa**
Falta de segmentação	Criar VLANs + Regras básicas de firewall	Reduz movimento lateral e aumenta isolamento de redes
Portas abertas críticas	Fechar serviços não utilizados (21, 139, 3306, 389, 445)	Minimiza superfície de ataque
Falta de firewall interno	Implementar políticas de firewall nas interfaces com iptables ou ufw	Impede comunicação desnecessária entre redes
Falta de monitoramento	Implantar monitoramento com Zabbix e alertas para portas críticas	Detecta varreduras, acessos indevidos e falhas
Serviços desatualizados	Atualizar MySQL (>= 8.4.5), revisar versões LDAP/FTP	Reduz risco de exploração por vulnerabilidades conhecidas
LDAP aberto	Desabilitar acesso anônimo + aplicar autenticação forte (LDAPS com certificado)	Protege informações sensíveis expostas via bind anônimo
Inventário e backup	Consolidar ativos com script de rede e manter backups atualizados	Facilita rastreabilidade e auditorias futuras
Porta 55699	Identificar e justificar o serviço; se não crítico, desativar	Porta aparece em múltiplas redes, potencial risco oculto

Conclusão

O projeto entregou uma visão abrangente do ambiente de rede, evidenciando tanto os pontos fortes quanto os aspectos que demandam atenção. As recomendações apresentadas, junto ao plano de ação proposto, oferecem uma base consistente para a evolução contínua da segurança e a proteção proativa da infraestrutura.

Referências

https://www.google.com/?hl=pt_BR

https://chatgpt.com/

https://www.cec.health.nsw.gov.au/CEC-Academy/quality-improvement-tools/pareto-charts