UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIENCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTAÇAO E SISTEMAS

Jhonny Oliveira da Silva

ANÁLISE E TESTES DE INTRUSÃO EM DISPOSITIVOS IOT

João Monlevade
Ano 2023

JHONNY OLIVEIRA DA SILVA

ANÁLISE E TESTES DE INTRUSÃO EM DISPOSITIVOS IOT

Monografia apresentada ao curso Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Orientador: Dr. Theo Silva Lins

João Monlevade

Ano 2023

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTACAO E SISTEMAS

FOLHA DE APROVAÇÃO

Jhonny Oliveira da Silva

Análise e Testes de Intrusão em Dispositivos IoT

Monografia apresentada ao Curso de Engenharia de Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharelado em Engenharia de Computação

Aprovada em 30 de março de 2023

Membros da banca

Doutor - Theo Silva Lins - Orientador(a) Universidade Federal de Ouro Preto
Doutor - Marlon Paolo Lima - Universidade Federal de Ouro Preto
Doutor - Carlos Henrique Gomes Ferreira - Universidade Federal de Ouro Preto

Theo Silva Lins, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 12/04/2023

Documento assinado eletronicamente por **Theo Silva Lins**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 13/04/2023, às 13:47, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0508476** e o código CRC **C01425B7**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: (31)3808-0819 - www.ufop.br

DEDICATÓRIA

Esta dedicatória é para todas as pessoas que desejam compreender como elas podem estar vulneráveis a ataques de cibercriminosos, com o objetivo de aumentar sua consciência sobre o que não fazer e como se proteger contra tais ataques.

AGRADECIMENTOS

Eu agradeço a Deus por guiar meu caminho, proporcionar-me saúde e força para seguir em frente. Também sou grato à minha família, especialmente à minha irmã Verônica e à minha mãe Rosilaine, que sempre esteve presente e me apoiou ao longo dessa jornada, sendo um exemplo de mulher trabalhadora. Agradeço aos meus professores pelo conhecimento compartilhado ao longo da trajetória, em especial ao meu orientador Theo, sem o qual este trabalho provavelmente não teria sido concluído. Aos meus amigos e colegas, que estiveram presentes em todos esses anos de muito aprendizado em uma cidade nova para mim, agradeço pelo apoio e pelas boas conversas durante essa jornada. Por fim, sou grato à minha companheira Vitória por ter me apoiado nos momentos difíceis.

"Learn the rules like a pro so you can break them like an artist."

(Pablo Picasso)

RESUMO

Este estudo aborda a ascensão tecnológica da IoT (Internet das coisas) e sua crescente automatização em diversos setores, incluindo a vida social, através da comunicação máquina-máquina. Com o aumento de dispositivos e sensores inteligentes conectados à internet, surge a necessidade de conscientização sobre a relevância da segurança da informação na IoT. O objetivo do estudo é analisar e identificar as principais vulnerabilidades em sistemas IoT, bem como realizar testes de intrusão em dispositivos IoT para compreender como estes dispositivos são afetados em um cenário real e como se proteger. Será montado um ambiente de testes, com diversos dispositivos IoT presentes em uma residência, para ilustrar um ambiente de usuário doméstico. Será utilizado o sistema Linux e o Windows para manusear as ferramentas necessárias e providenciar os ataques. A justificativa do estudo é a popularização da Internet pelo uso de aplicativos em smartphones, smart Tv's e computadores, que levaram a um grande aumento de dispositivos inteligentes conectados à internet e, consequentemente, a uma maior exposição das pessoas a ataques de usuários mal-intencionados na internet.

Palavras-chave: IoT, segurança da informação, testes de intrusão, vulnerabilidades, Pentest.

ABSTRACT

This study addresses the technological rise of IoT (Internet of Things) and its increasing automation in various sectors, including social life, through machine-to-machine communication. With the increase of smart devices and sensors connected to the internet, there is a need for awareness about the relevance of information security in IoT. The aim of the study is to analyze and identify the main vulnerabilities in IoT systems, as well as conduct intrusion tests on IoT devices to understand how these devices are affected in a real scenario and how to protect them. A test environment will be set up, with various IoT devices present in a household, to illustrate a home user environment. Linux and Windows systems will be used to handle the necessary tools and provide the attacks. The justification for the study is the popularization of the Internet through the use of applications on smartphones, smart TVs, and computers, which has led to a significant increase in smart devices connected to the Internet and, consequently, greater exposure of people to attacks by malicious users on the Internet.

Keywords: IoT, information security, intrusion testing, vulnerabilities, Pentest.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS

FTP – File Transfer Protocol

OWASP – Open Web Application Security Project

SSH – Secure Shell

IoT – Internet of Things

TCP – Transmission Control Protocol

DNS – Domain Name System

WiFi – Wireless Fidelity

Telnet – Telecommunication Network

IP – Internet Protocol

MAC – Media Access Control

SNMP – Simple Network Management Protocol

ARP – Address Resolution Protocol

JS – Javascript

XSS – Cross-Site Scripting

LAN – Local Area Network

PoC - Proof of Concept

SUMÁRIO

1 Introdução

A ascensão tecnológica, tende a transformar todas as estruturas da sociedade, impactando, com a crescente evolução, inclusive, os âmbito econômicos e sociais propriamente ditas, ou seja, na relação entre indivíduos. Assim, a IoT (Internet das coisas) definida, por Magrani (2018) como a crescente automatização de diversos setores, desde a economia até a vida social através da comunicação sem a intervenção do ser humano, ou seja, máquina-máquina. E de acordo com Kranenburg et al (2011), ao afirmar que a internet é cada vez mais coletiva, e que tudo, incluindo artefatos físicos estão ou estarão conectados. Assim, para que o progresso seja benéfico, faz-se necessário que a segurança e a privacidade dos usuários se expandam na mesma medida, já que, conforme Magrani (2018), o aumento de usuários em detrimento aos ultra-processadores fazem com que os primeiros estejam mais suscetíveis aos ataques cibernéticos.

O trabalho remoto em home office se tornou mais comum devido à pandemia do novo coronavírus, que levou as empresas a adaptarem suas estratégias. De acordo com uma pesquisa da FGV, 33% das empresas no Brasil adotaram o home office (FOLHA DE S.PAULO, 2023). Nesse contexto, os autores Toso e Pereira Júnior (2021) destacam que a ampla adoção do home office pode enfraquecer a segurança das redes corporativas, uma vez que estende seu perímetro até as residências e compromete as políticas de segurança planejadas para ambientes operacionais distintos. Frequentemente, quando as pessoas adotam o trabalho remoto, elas podem não receber treinamento adequado sobre as medidas de segurança necessárias para proteger suas informações e garantir que a sua rede esteja segura. Além disso, é importante ressaltar a importância da segurança das informações em dispositivos de uso diário, uma vez que os fabricantes estão incluindo sensores inteligentes em uma ampla variedade de equipamentos usados no cotidiano, visando proporcionar conforto, praticidade e conectividade.

1.1 Problema

Conforme estudos realizados pela (*IoT Analytics*, 2022), já existem mais de 14,4 bilhões de dispositivos conectados. Além disso, a pesquisa também indica que o setor apresentou retomada de expansão no ano em questão. Projeções apontam que

até 2025 esse número pode chegar a aproximadamente 27 bilhões de dispositivos conectados.

Além disso, o aumento do número de pessoas adotando o home office e utilizando a Internet, inclusive para fins de trabalho, é fundamental conscientizar sobre a importância da segurança da informação e das medidas necessárias para garantir a proteção dos dados. Isso é especialmente crucial em dispositivos IoT, que possuem sensores que permitem a comunicação com outros sistemas, tornando-os mais vulneráveis a fraudes e roubo de dados.

Diante disso, mostra-se imprescindível, o estudo e observação das principais formas de ataques aos dispositivos, a fim de compreender seu funcionamento e, posteriormente desenvolver soluções com o propósito de preservar a tríade CIA (do inglês Confidentiality, Integrity and Availability), e garantir a segurança do ambiente no qual estes dispositivos são inseridos.

1.2 Objetivos

Realizar uma análise abrangente dos dispositivos IoT, identificando possíveis vulnerabilidades e pontos de entrada para ataques cibernéticos, bem como desenvolver e propor medidas de segurança eficazes para garantir a integridade, confidencialidade e disponibilidade dos dados e informações armazenados nos dispositivos em uma rede residencial. Isso será alcançado por meio de testes de intrusão em dispositivos IoT, a fim de compreender como esses dispositivos são afetados em um cenário real e como se proteger contra possíveis ameaças.

1.2.1 Objetivos Específicos

- Realizar uma análise minuciosa de dispositivos IoT, a fim de identificar suas vulnerabilidades e possíveis pontos de entrada para intrusões.
- Identificar as formas mais comuns de ataques cibernéticos que podem comprometer a integridade, confidencialidade e disponibilidade dos dispositivos IoT em um cenário real.
- Realizar testes de intrusão em dispositivos IoT para simular possíveis ataques cibernéticos e avaliar a eficácia das medidas de segurança utilizadas.

- Compreender como os dispositivos IoT são afetados em um cenário real, a fim de obter uma visão mais clara dos riscos envolvidos na utilização desses dispositivos.
- Propor medidas de segurança eficazes para proteger os dispositivos IoT em uma rede residencial, a fim de garantir a integridade, confidencialidade e disponibilidade dos dados e informações armazenados nos dispositivos.

1.3 Justificativa

O surgimento das redes de Internet da Coisas e a popularização da Internet pelo uso de aplicativos em smartphones, smart Tv e computadores, levaram a um grande aumento de dispositivos inteligentes conectados à Internet.

Por conta disso, pessoas comuns que muitas vezes tem pouco conhecimento sobre segurança da informação, estão cada vez mais expostas a ataques de usuários mal intencionados na internet.

1.4 Metodologia

De acordo com a metodologia adotada neste estudo, foi realizado um levantamento das principais vulnerabilidades encontradas em sistemas IoT segundo a OWASP, selecionando as cinco melhores ranqueadas para estudar técnicas de ataque e como elas são exploradas em um cenário real. Em seguida, foi montado e configurado um ambiente de testes com diversos dispositivos IoT presentes em uma residência, ilustrando um ambiente de usuário doméstico. O estudo consiste em técnicas utilizadas por Pentest para realizar testes de penetração, com o objetivo de aplicá-las nos testes deste laboratório. Para realizar esses testes, foram utilizados o sistemas Linux Parot e o Windows para manusear as ferramentas necessárias e providenciar os testes. O computador utilizado para os testes possui 8GB RAM(DDR4), Placa de vídeo onboard Intel UHD Graphics 620 e processador Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz.

2 Referencial Teórico

Neste capítulo o objetivo é mostrar todos os conceitos que foram usados como base para a pesquisa. E para isso, foram abordados tópicos sobre Owasp, IoT, problemas de segurança encontrados nestes dispositivos, ferramentas utilizadas por pentesters e os tipos de ataques utilizados.

2.1 Owasp

É uma organização sem fins lucrativos que tem como objetivo auxiliar as pessoas a desenvolverem softwares seguros. A OWASP possui um projeto com foco em chamado OWASP IoT Top 10, na qual possui uma lista contendo as 10 vulnerabilidades de segurança cibernética mais comuns encontradas em dispositivos e sistemas IoT, essa lista foi desenvolvida como uma ferramenta para ajudar as organizações a identificar e mitigar os riscos de segurança mais críticos relacionados ao IoT.

Dessas 10 vulnerabilidades, foram abordadas as cinco principais durante os testes.

2.1.1 Interfaces Web Inseguras

Isso inclui a falta de medidas de segurança básicas em dispositivos IoT, criptografia insuficiente, falta de autenticação.

2.1.2 Credenciais fracas e senhas padrões

Muitos dispositivos IoT vêm com senhas padrão pré-configuradas que são fáceis de adivinhar, ou permitem que as credenciais sejam armazenadas de forma insegura.

2.1.3 Serviços de rede inseguros

São serviços que não oferecem medidas de segurança adequadas para proteger os dados transmitidos ou armazenados. Ou podem estar desatualizados contendo alguma vulnerabilidade já publicada.

2.1.4 Falta de criptografia durante o transporte de informações

Dispositivos IoT frequentemente se comunicam entre si e com outros dispositivos através de conexões inseguras sem criptografia, como Wi-Fi ou Bluetooth, deixando-os vulneráveis a interceptação de dados.

2.1.5 Falta de privacidade do usuário

Muitos dispositivos IoT coletam e armazenam dados pessoais dos usuários sem sua permissão ou sem fornecer informações claras sobre como esses dados serão usados.

2.2 Ferramentas Utilizadas

Pentest é um processo utilizado para avaliar a segurança de sistemas ou redes, simulando um ataque cibernético real, com o objetivo de identificar vulnerabilidades antes que possam ser exploradas por invasores. A utilização de ferramentas é essencial para a execução de um pentest, sendo que cada uma possui um propósito específico. Nesta seção, serão abordadas as ferramentas que serão utilizadas durante os testes.

2.2.1 Nmap

O Nmap é um programa de escaneamento de rede open-source que é amplamente utilizada para descobrir dispositivos e serviços em uma rede. Alguns dispositivos possuem dezenas de portas abertas (Loi et al., 2017) que expõem serviços como SSH ou Telnet.

Algumas das suas principais funcionalidades incluem:

- Escaneamento de endereços IP e portas: Nmap pode varrer uma rede para identificar quais endereços IP e portas estão ativos e disponíveis.
- Identificação de sistemas operacionais: Nmap pode analisar pacotes de rede para identificar o sistema operacional em uso em um dispositivo específico.
- Detecção de serviços: Nmap pode identificar quais serviços estão sendo executados em uma porta específica, como HTTP, FTP, SSH, entre outros.

- Escaneamento personalizado: Nmap permite a configuração de escaneamentos personalizados, como varreduras lentas, varreduras silenciosas e varreduras de portas específicas.

Esta ferramenta é amplamente usada para descobrir e gerenciar dispositivos e serviços em uma rede, identificar vulnerabilidades de segurança e planejar melhorias para a segurança da rede.

2.2.2 Ferramentas do Desenvolvedor (Navegadores de Internet)

Essas ferramentas de desenvolvedores presentes nos navegadores populares permitem rapidamente o conteúdo/recursos de uma página da Web. Elas incluem recursos como depurador de código, visualizador de elemento, console de JavaScript e ferramentas de rede.

2.2.3 Hydra THC

O THC Hydra é uma ferramenta que usa dicionário de força bruta para ataques e tentativas de várias combinações de senhas e logins contra um alvo.

Algumas das principais funcionalidades do THC Hydra incluem:

- Suporte para vários protocolos como smb, sip, imap, telnet, ftp ,ssh, http, entre outros.
- Suporte a vários tipos de autenticação: O THC Hydra suporta vários tipos de autenticação, incluindo autenticação básica, digest, NTLM e Kerberos.
- Suporte a paralelismo: O THC Hydra suporta a execução de vários testes de senha simultaneamente, o que aumenta a velocidade de teste.
- Suporte a wordlist e regra personalizada: O THC Hydra permite que o usuário use sua própria lista de palavras-chave ou regra para gerar as tentativas de senha.

2.2.4 WinSpy++

WinSpy++ é uma ferramenta do programador usada para "espiar" as janelas de um aplicativo e permite que você visualize e até modifique algumas de suas propriedades.

2.2.5 Wireshark

Como alguns dispositivos IoT se, comunicam de maneira diferente com base em seu estado, é importante capturar o tráfego de rede dos dispositivos em diferentes estados (Pesce, 2017):

- Iniciando, sem configuração e em stand.
- Comunicação com o aplicativo móvel e web e desktop.
- Durante uma atualização de firmware;
- Sem conexão com a internet.

Portanto, para captura do tráfego de rede, utilizaremos o Wireshark. Ele é uma ferramenta de análise de rede open-source que permite aos usuários visualizar e analisar o tráfego de rede em tempo real. Wireshark é amplamente utilizado por administradores de redes, desenvolvedores vos, engenheiros de segurança e outros profissionais que precisam monitorar, analisar o tráfego de rede e detectar e corrigir problemas.

Algumas das principais funcionalidades do Wireshark incluem:

- Captura de pacotes: O Wireshark pode capturar pacotes de rede em tempo real, permitindo que os usuários vejam o que está acontecendo em sua rede.
- Análise de pacotes: O Wireshark permite aos usuários visualizar e analisar pacotes capturados, exibindo informações como endereços IP, portas, protocolos e dados contidos nos pacotes.
- Filtros de pacotes: O Wireshark permite que os usuários apliquem filtros para visualizar pacotes específicos ou para ocultar pacotes indesejados.
- Suporte a diversos protocolos: Wireshark suporta uma ampla variedade de protocolos, incluindo Ethernet, TCP, UDP, HTTP, DNS, entre outros

- Ferramentas de análise: Wireshark fornece ferramentas para ajudar na análise de tráfego de rede, como gráficos estatísticos, conversas e fluxos.

2.2.6 DIRB

DIRB é uma ferramenta de escaneamento de vulnerabilidades de aplicativos web que é usada para descobrir arquivos e diretórios escondidos em um servidor web. O DIRB funciona tentando acessar uma série de URLs com nomes de arquivos e diretórios comuns, com base em uma *Wordlist* pré-configurada ou personalizada, sendo capaz de detectar arquivos e diretórios que podem ser acessados por meio de diretórios existentes ou de links de redirecionamento.

2.2.7 Jadx Decompiler

O Jadx foi projetado para descompilar arquivos de bytecode do Android, tornando-os legíveis e editáveis para os desenvolvedores. Com uma interface gráfica simples e intuitiva, os usuários podem importar arquivos APK ou diretórios de aplicativos, além de obter informações detalhadas sobre as classes e métodos que estão sendo descompilados. Além disso, o Jadx Decompiler também suporta recursos avançados, como a análise de código e a exibição de informações sobre as chamadas de API.

2.2.8 Dnsmasq

O Dnsmasq é uma ferramenta de software livre e de código aberto que combina as funções de um servidor DHCP e de um servidor DNS. Uma vantagem do dnsmasq baixa utilização de recursos de sistema. O Dnsmasq pode ser executado em hardware de baixo custo, como roteadores domésticos, sem afetar significativamente o desempenho do sistema.

2.2.9 Winbox

Winbox é um software de gerenciamento de rede utilizado para configurar e administrar dispositivos MikroTik RouterOS. Com o Winbox, os usuários podem acessar uma variedade de recursos, incluindo a configuração de interfaces de rede, o

gerenciamento de rotas, o controle de firewall, o gerenciamento de usuários e grupos e a configuração de serviços de rede, como DNS e DHCP.

O Winbox utiliza a porta TCP 8291 para comunicação com os dispositivos MikroTik RouterOS. É importante garantir que essa porta esteja aberta no firewall do dispositivo e que somente os endereços IP autorizados tenham acesso a ela para garantir a segurança da rede. Além disso, é possível configurar uma porta personalizada para o Winbox em vez da porta padrão 8291, se necessário.

3 Trabalhos Relacionados

A fim de desenvolver o presente trabalho, foram abordados problemas de segurança dos dispositivos IoT, em especial, aqueles previstos pela OWASP, que, no que lhe concerne, de acordo com (LEITE, 2019) corresponde à:

> (...) uma comunidade online que cria e disponibiliza de forma gratuita artigos, metodologias, documentação, ferramentas e tecnologias no campo da segurança de aplicações web (OWASP, 2018). Com o objetivo de ajudar fabricantes, desenvolvedores e consumidores a entender melhor as questões de segurança associadas à IoT, surgiu o projeto OWASP Internet of Things, permitindo que usuários, em qualquer contexto, tomem as melhores decisões de segurança ao criar, implantar e avaliar tecnologias de IoT (OWASP, 2018).

Tão logo, as citadas vulnerabilidades foram listadas, respectivamente como: senhas fracas, previsíveis ou dentro do código; Serviços de rede inseguros; Ecossistema de interfaces inseguros; Falta de mecanismos de atualização seguros; Uso de componentes inseguros ou obsoletos; Proteção da privacidade insuficiente; Transferência e armazenamento de dados de maneira insegura; Falta de controle de gerenciamento dos dispositivos; Configuração insegura por padrão; Segurança física insuficiente.

Nesse contexto, destaca-se os requisitos de segurança mais importantes, listados por (LEITE, 2019), definidos tais como a Tríade da Segurança de Informação: confidencialidade, integridade e disponibilidade, estes, no que lhes dizem respeito são aplicáveis a qualquer tipo de rede (MOSENIA e JHA, 2016).

Ocorre que as redes estão, no que lhes concernem suscetíveis a diversos tipos de ataques, classificados tais por Andrea, Chrysostomou e Hadjichristofi (2015) como físicos, nos quais o invasor precisa ter acesso físico aos dispositivos, sendo incluídos ainda, os ataques que prejudicam a vida útil ou o funcionamento do hardware (LEITE, 2019), de rede, de software, os quais, exploram o sistema usando trojans, worms, vírus, spyware e scripts maliciosos que podem roubar informações, adulterar dados, negar serviço e até danificar os dispositivos de um sistema de IoT (LEITE, 2019), e, finalmente, de criptografia.

Então, a fim de promover a proteção da rede, buscou-se o desenvolvimento de um laboratório com diversos dispositivos IoT e, a partir da realização de testes de

intrusão nos laboratórios criados, visa-se o desenvolvimento de soluções para as vulnerabilidades encontradas.

Nesse âmbito, destacam-se os artigos, utilizados tais para desenvolvimento do presente trabalho como fonte bibliográfica, o artigo “Análise de requisitos de segurança para uma rede de IoT”, (Monteiro, 2020) destacou a importância da análise de requisitos de segurança em redes de IoT residenciais e de pequeno porte, a fim de garantir a proteção dessas redes. Segundo o autor, também é necessário apresentar as definições e padrões de senhas para a utilização de dispositivos em redes IoT. Inclusive, foi utilizado também o artigo “Internet das Coisas (IoT): Vulnerabilidades de Segurança e Desafios”, de Leite (2021, p. 10), que tratou a temática identificando as vulnerabilidades, expondo casos notórios e propondo possíveis soluções para a segurança do ambiente.

Além disso, o artigo de Berlanda, (2020) “Guia de segurança da informação para a conectividade de dispositivos IoT” visou o desenvolvimento de um guia de segurança da informação para dispositivos IoT utilizados na Indústria 4.0 através da análise e classificação das principais falhas e riscos em segurança da informação e, finalmente, o artigo “Segurança da Informação em IoT” de Fukada, (2021) objetivou o mapeamento de vulnerabilidades e a reflexão sobre indicadores de Segurança de Informação, realizando critérios de segurança de informação em ambientes de conexão IoT.

Segue tabela 1 com comparativo entre os principais artigos e suas formas de abordagem do tema:

Tabela 1 - Quadro Comparativo

	Análise e testes de intrusão em dispositivos IoT	**(Monteiro, 2021)**	**(Leite, 2019)**	**(Berlanda, 2021)**	**(Fukada, 2019)**
Tipo de pesquisa	Pesquisa bibliográfica	Pesquisa bibliográfica	Pesquisa bibliográfica	Pesquisa bibliográfica	Pesquisa bibliográfica
Tema	Segurança de redes IoT, focada em redes residenciais.	Segurança em redes IoT residenciais e de pequeno porte	Identificação de vulnerabilidades, ameaças e tipos de ataque	Desenvolvimento de guia de segurança de equipamentos da Indústria 4.0	Entendimento dos dispositivos de Segurança de informação em dispositivos de conexão
Metodologia	Mapeamento de vulnerabilidades	Requisitos de segurança das senhas para os equipamentos de roteamento	Exposição de casos	Identificação de riscos	Mapeamento de vulnerabilidades
Resultados	Realização de experimento	Definições e padrões de senhas a ser utilizadas	Proposta de possíveis soluções para segurança do ambiente IoT	Realização de experimento	Vantagens da IoT
Considerações finais	Recomendações			Recomendações	Análise de critérios de segurança

4 Desenvolvimento e Experimentos

Neste capítulo será apresentada uma análise dos dados coletados, a partir da investigação realizada e por meio da metodologia aplicada.

4.1 Cenário do Experimento

Para o desenvolvimento da pesquisa e construção do experimento foram utilizados os seguintes dispositivos:

1. Par de roteadores Mesh
2. ONU dualband
3. Camêra IP
4. Sensor de umidade para solos, desenvolvido com o microcontrolador ESP8266.
5. Smart TV
6. Google Nest Mini
7. 2 Interruptores Wi-Fi, de marcas distintas.
8. Roteador Microtik
9. Impresora

A Figura 1 apresenta uma visão aérea que permite uma visualização mais clara da disposição dos dispositivos.

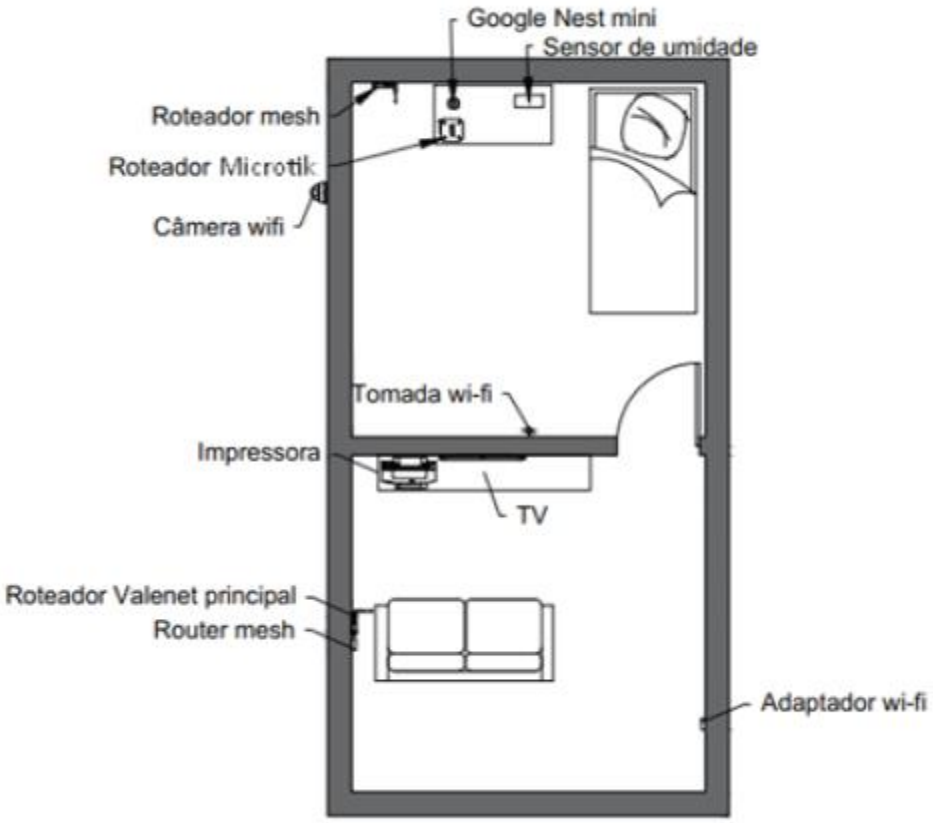

Figura 1 - Vista superior do laboratório

4.2 Resultados dos Experimentos Realizados

Nesta seção serão apresentados alguns testes realizados nos dispositivos e no fim serão relacionados com as principais vulnerabilidades citadas do top 10 OWASP.

4.2.1 Par de roteadores Mesh

Esse par de roteadores mesh da marca Tenda, são equipamentos muito utilizados por empresas do ramo de telecomunicações.

Na Figura 2 é mostrado o Mesh que é um aparelho que amplia o alcance do Wi-Fi e distribui o sinal nos ambientes de forma fácil e rápida. Essa tecnologia conecta os dispositivos à internet sem perder qualidade. Explicando de uma maneira fácil, o roteador Mesh cria uma rede inteligente porque faz a conexão do aparelho ao melhor ponto do sinal Wi-Fi automaticamente, ou seja, os próprios módulos mesh controlam o uso de frequências e canais de forma transparente, sendo possível aproveitar sempre a melhor capacidade da rede sem precisar de alterações manuais.

Figura 2 - Roteadores Mesh Tenda Wi-Fi

Para primeira etapa a de reconhecimento foi utilizado o Nmap, onde é possível realizar escaneamento de portas e detectar possíveis locais que possam ser explorados.

Foi utilizado dois argumentos -sT (TCP scan) que é utilizado para tentar realizar um three-way-handshake em cada porta. Se a conexão for bem sucedida, conclui-se que a porta está aberta. Apesar de ser uma varredura fácil de ser detectada, ela é a que apresenta resultados mais confiáveis. Na Figura 3, foi observado o -p-

(Equivalente ao -p1-65535), para escanear todas as portas TCP disponíveis, pois por padrão o nmap escaneia as primeiras 1000 portas.

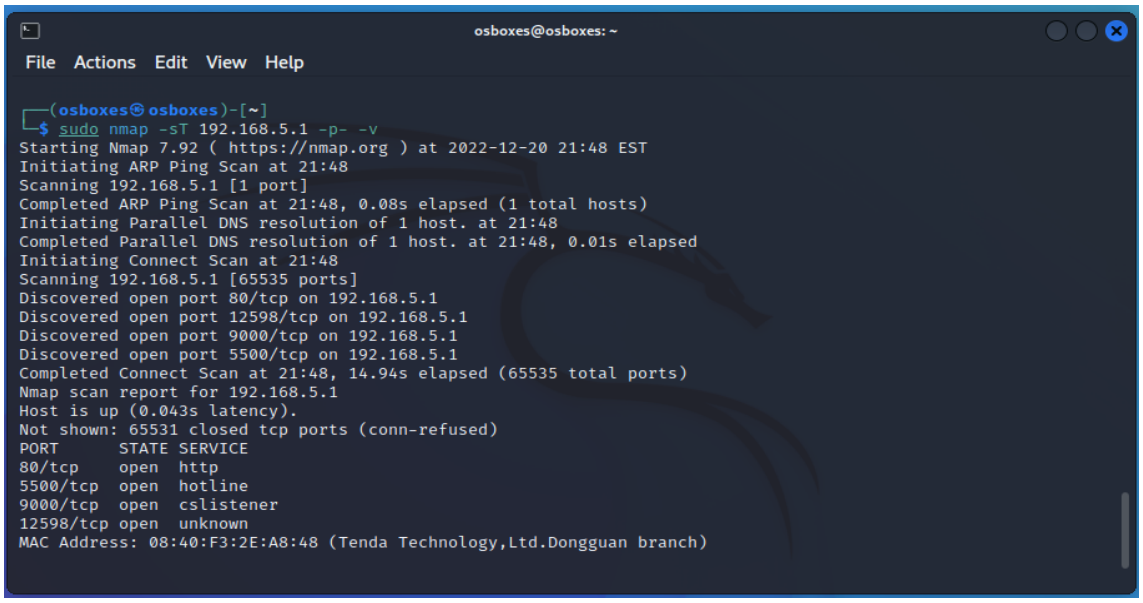

Figura 3 - Varredura com o nmap

Para ter acesso a porta 80 que utiliza protocolo http, basta digitar o IP no navegador e abrirá a sua interface mostrada na Figura 4. O primeiro passo para ser explorado é a aplicação e conhecida na sua codificação, portanto utiliza-se a ferramenta do próprio chrome “DevTools” – Ferramentas de desenvolvedor.

Com esta ferramenta é possível analisar requisições, código fonte (apenas o que roda do lado do cliente), debugar e etc.

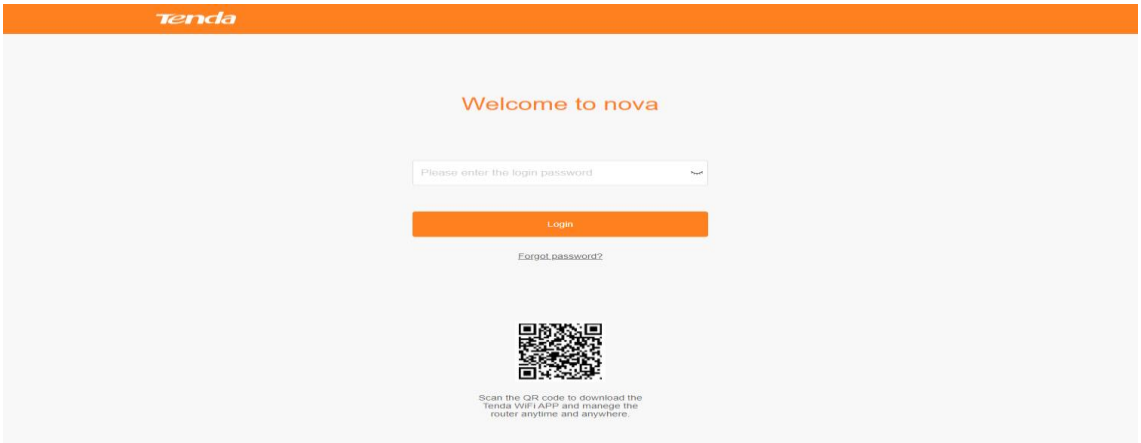

Figura 4 - Interface Web Mesh

Acessando a interface de administração do roteador e inspecionando o html, nota-se que existem poucas opções de configurações, fabricantes fazem isso para limitar o acesso a opções que possam causar danos se mal configuradas ou se configuradas por pessoas mau intencionadas.

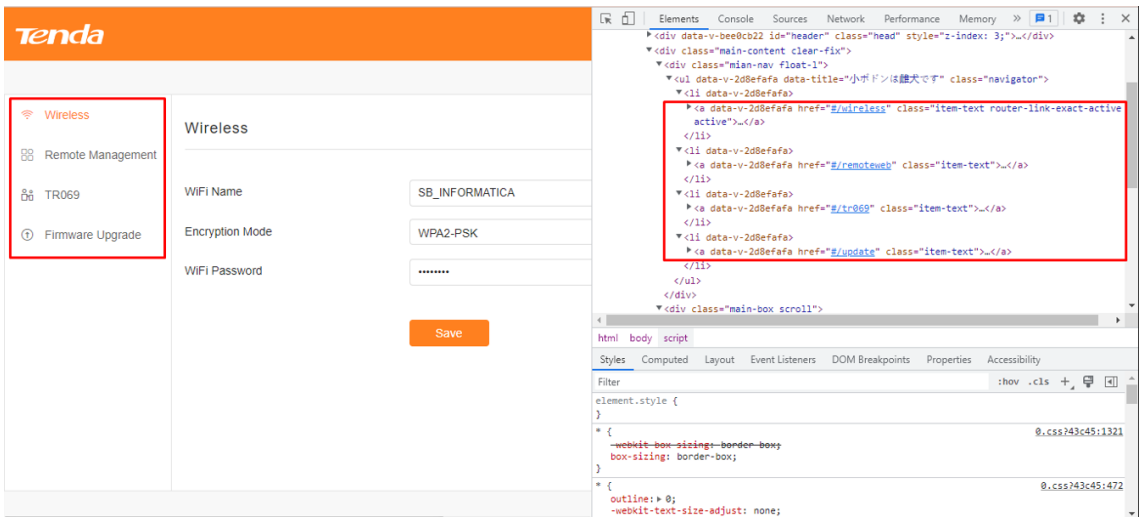

Figura 5 - Inspeção dos elementos da interface web

O problema apresentado na Figura 5 é que o caminho das páginas pode ser encontrado no próprio JavaScript como pode ser visto na imagem, portanto basta apenas copiar o caminho e acessar a página pretendida.

A página aberta na Figura 6, é um dos principais alvos de cibercriminosos, que é onde é possível alterar o seu DNS, atacantes alteram esse campo para servidores DNS controlados por eles, no intuito de redirecionar o usuário para páginas com conteúdo malicioso.

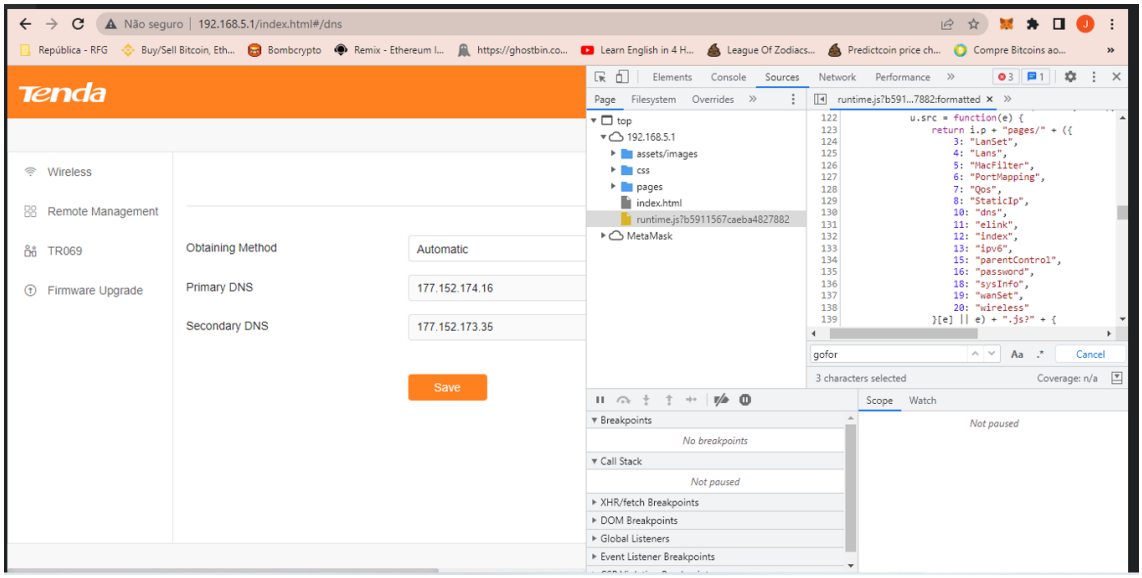

Figura 6 – Código JavaScript mostrando páginas presentes no servidor Web

Outro ponto a ser observado do DevTools é a aba “Network” mostrado na figura 7, nela é possível ver pacotes de rede que são trafegados a cada requisição na página. Após logar na interface do roteador podemos ver três requisições get?module_id= ****, elas são utilizadas para retornar informações de configuração do roteador para a interface web.

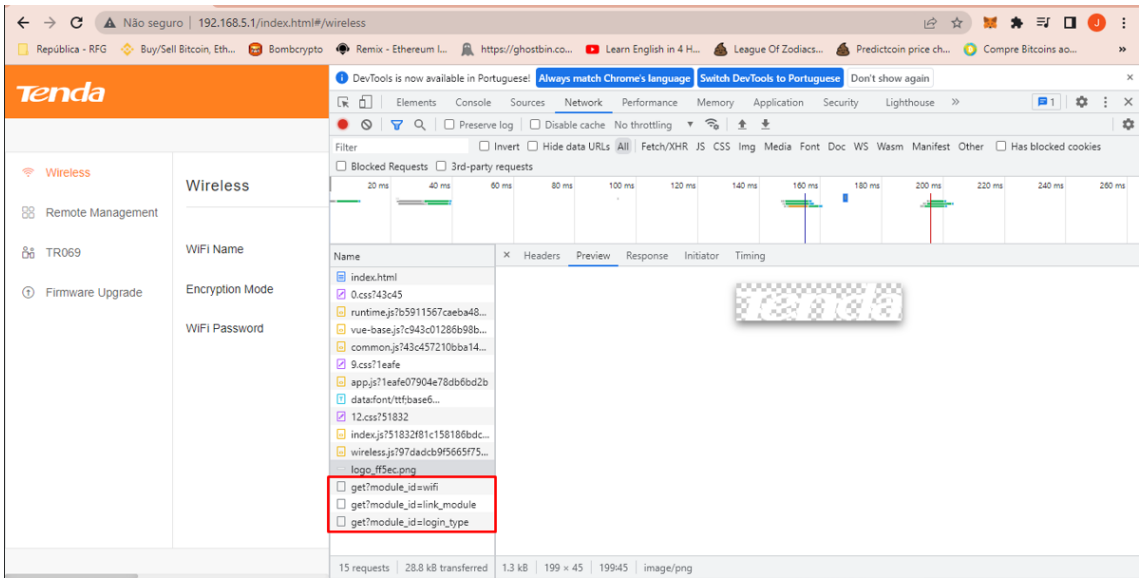

Figura 7 - Requisições realizadas durante o uso da aplicação

O problema aqui é que não é feito nenhuma validação de autenticação ao realizar esses “GET”, o que significa que mesmo sem estar logado ao roteador,

qualquer pessoa que consiga enxergar o IP deste roteador, ou seja, qualquer dispositivo conectado nesta rede ou em redes subjacentes que possua rotas para se chegar até ele, conseguiria realizar essas requisições. Um exemplo pode ser visto logo abaixo na Figura 8, onde é-se retornado informações da rede Wi-FI, login e senha mesmo não estando logado na interface de administração do roteador.

Fazendo uma varredura pelas requisições e mesclando com as páginas ocultas descobertas podemos encontrar os seguintes argumentos funcionais para o get?module_id= “ *** ” :

wifi - (Retorna informações da rede wi-fi), ilustrado na Figura 8.

login_type - (Permissões de usuário)

port_list,lan_info - (Informações da rede Lan, e portas do roteador)

tr069 - (Informações sobre o protocolo, que expõe inclusive informações sobre a infraestrutura do provedor)

link_module,static_wan_info,dynamic_wan_info,pppoe_cfg,mac_clone – (Informações da Wan)

login_pwd - (Informações sobre Login da Interface Web – que é o que nós queremos)

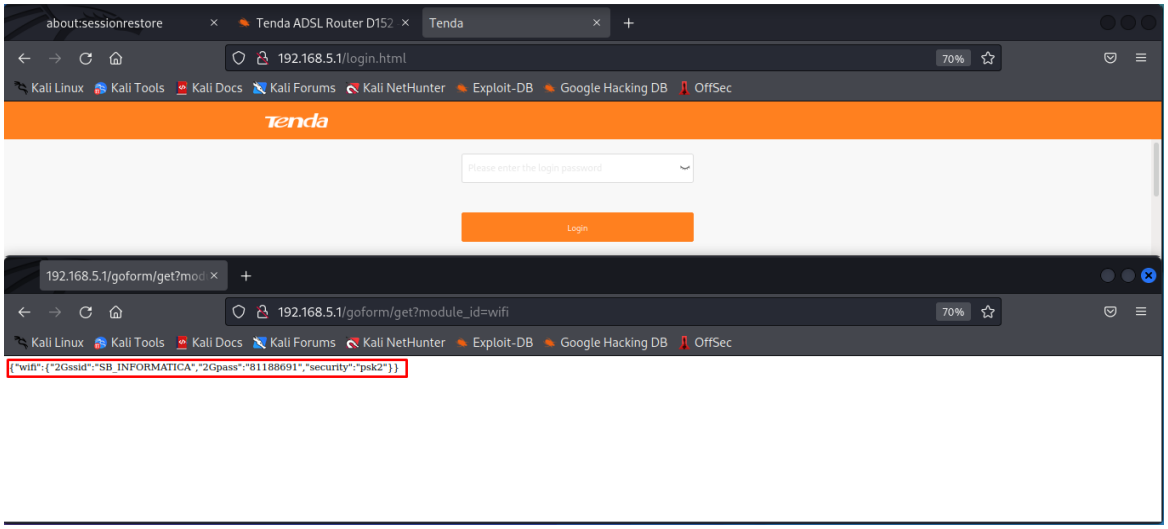

Figura 8 - Utilizando o end-point de consultas para obter credenciais da rede wireless

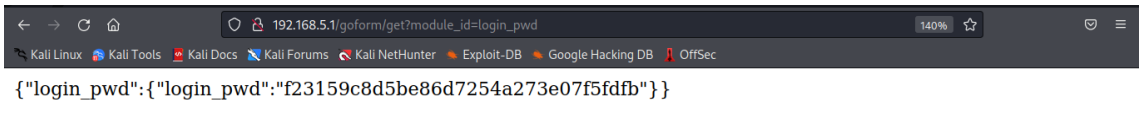

Figura 9 - Consulta a senha da interface de administração do roteador

Passando o argumento login_pwd, temos a saída mostrada na Figura 9, que me retorna uma informação criptografada. É importante salientar que, apesar de uma aparente sensação de segurança, existem sites que possuem bancos de dados extensos contendo senhas que foram previamente transformadas em hash. Essa situação pode facilitar a quebra de senhas que não possuem uma complexidade significativa, representando uma ameaça potencial à segurança. Essa constatação é ilustrada na Figura 10.

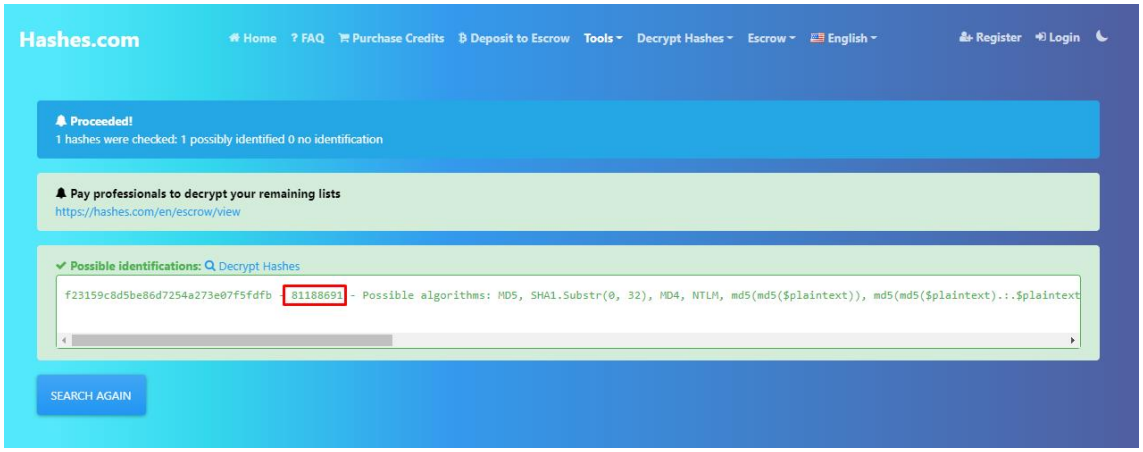

Figura 10 - Quebra de senhas criptografadas em md5

4.2.2 ONU dualband

Figura 11 - Roteador utilizado em uma empresa convencional da região

Diferente do exemplo anterior onde foi utilizando um Scan TCP 3-way handshake, neste exemplo será utilizado o -sS (TCP Syn Scan) que também é o modo de verificação padrão no Nmap, ela requer privilégio (root) para executá-la. A varredura SYN não precisa concluir o handshake tridirecional TCP, em vez disso, ele interrompe a conexão assim que recebe uma resposta do servidor. Como não estabelecemos uma conexão TCP, diminui as chances do escaneamento ser registrado nos logs do dispositivo.

Nota-se na Figura 12, que foram encontrados diversos serviços com status “Open”, ftp, ssh, Telnet, NetBios e em um cenário real essas portas abertas sem nenhum firewall para protege-las só vai aumentar as possibilidades do atacante conseguir comprometer o seu Sistema.

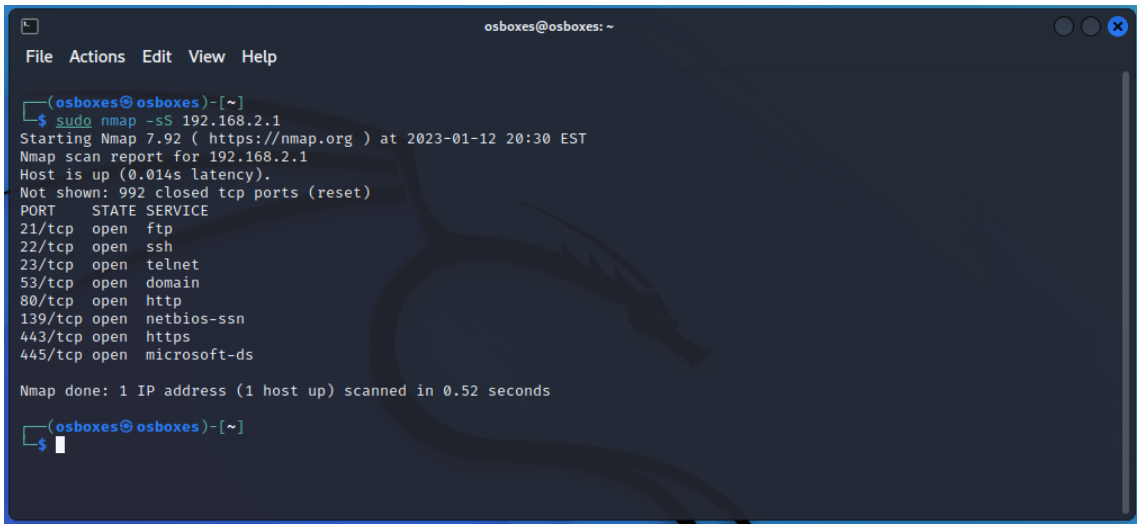

Figura 12 - Varredura de portas da onu com o Nmap

Para mostrar na prática, mostraremos nesta etapa o THC, ela é uma poderosa ferramenta para força bruta, capaz de realizar ataques de força bruta em diversos serviços tais como FTP, SSH, HTTP, Telnet, etc. Além do mais a ferramenta te dá opções de escolher a quantidade de threads durante os ataques, o que aumenta o desempenho durante a sua execução.

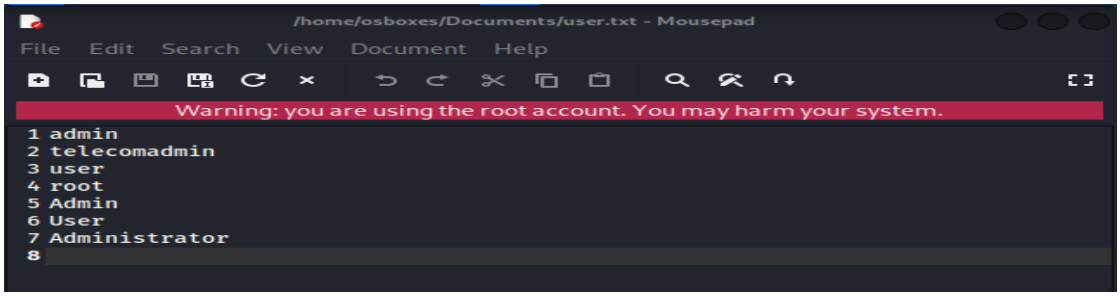

Figura 13 - Lista de usuários

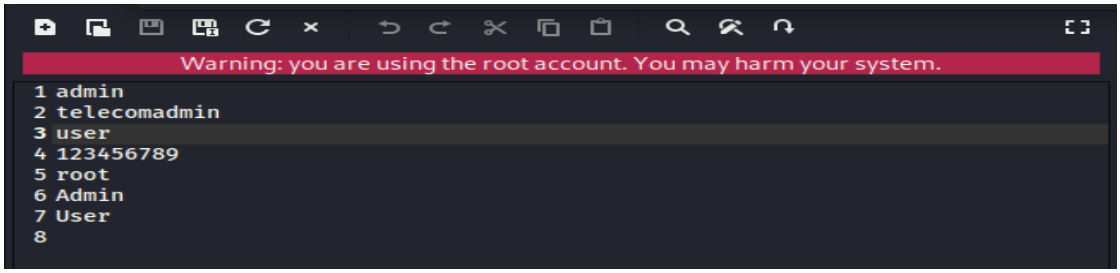

Figura 14 - Lista de senhas

Foi criado dois arquivos contendo algumas senhas e usuários padrões contidos em roteadores, essas serão as nossas duas WordList's, ambas podem ser vistas na Figura 13 e 14. As senhas padrões de alguns modelos de roteadores podem ser encontradas em Netspot (2022).

Agora podemos usar as wordlist's junto com o Hydra, vamos começar pelo Telnet, mostrado na Figura 15.

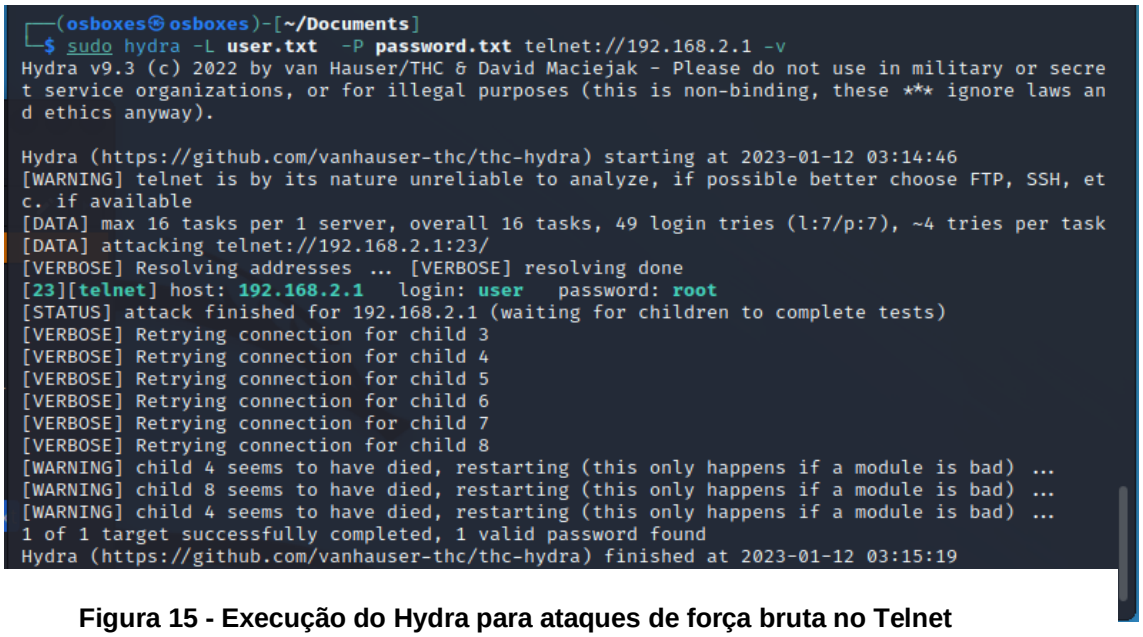

Figura 15 - Execução do Hydra para ataques de força bruta no Telnet

Podemos observar que ao todo foram testados 49 combinações, onde foi possível encontrar um usuário e senha válidos para o Telnet. A sintaxe do Hidra é a seguinte : -L (Argumento para se passar uma lista de usuários), -P(Passar uma lista de senhas), <serviço que você irá atacar>://<IP do dispositivo> e o -v (verbose) retorna informações sobre a execução na sua tela é um parâmetro opcional.

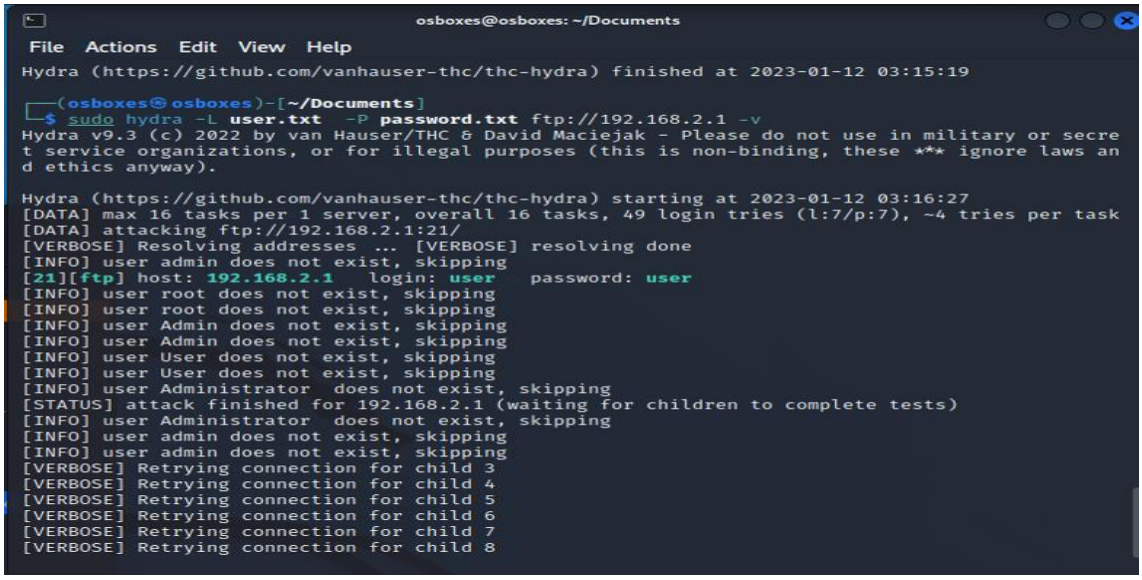

Figura 16 - Execução do Hydra para ataques de força bruta no ftp

Realizando o mesmo ataque com as mesmas *wordlists* no serviço de ftp mostrado Figura 16, obtemos mais um usuário válido que podemos usar para comprometer o dispositivo.

Utilizando as credencias descobertas conseguimos acessar ambos serviços, e para entender o impacto deste acesso não autorizado vamos entender primeiro um pouco sobre esses dois serviços.

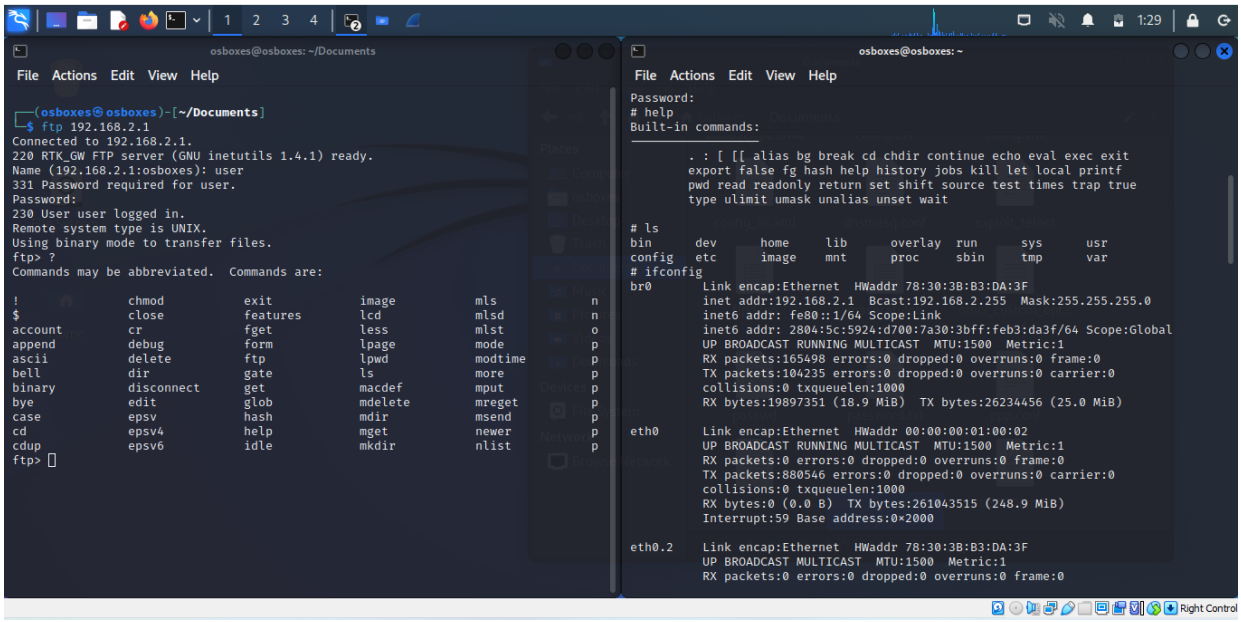

Figura 17 - Acessando o fpt e telnet com as credenciais descobertas

O FTP é um protocolo de modelo cliente servidor utilizado para enviar e receber arquivos, por uma rede de computadores, desde uma local à internet. Apesar de ser um protocolo antigo ele vem sendo bastante usado por sistemas de hospedagem de sites. No entanto, ele é considerado inseguro pois não conta com uma série de padrões de segurança suportados pelos navegadores de internet modernos.

Tendo acesso ao FTP do roteador temos a nossa disposição uma série de comando mostrados na Figura 17, com **get** é possível desde realizar download's de arquivos de configurações, arquivos sensíveis (passwd, log's) e o próprio código fonte da aplicação web rodando no roteador. Com o **put** ou **send**, é possível também enviar algum arquivo para o servidor caso o usuário tenha permissão de escrita, isso é útil pois a um atacante pois é fácil enviar um backdoor no dispositivo para garantir um acesso persistente, mesmo que o administrador mude a senha ou implemente regras de firewall.

O **Telnet** também é um protocolo antigo, ele é utilizado para acessar virtualmente um computador. Ele fornece um canal de comunicação entre duas máquinas via linhas de comando, permitindo um acesso remoto para executar funções ou ajudar na resolução de problemas. Isso implica que através do usuário descoberto conseguimos ter acesso a um sistema operacional inteiro com todas as bibliotecas e executáveis,

além de ser possível realizar instalações e modificações em arquivos, porém as restrições que você terá, vão depender das permissões do usuário.

Alguns exemplos de arquivos interessantes que foram encontrados, pode ser visto na Figura 18, incluí senha do wi-fi, ppoe, arquivo boa.conf que tem todas as configurações do servidor web inclusive autenticação, passwd (Criptografado com um salt), logs de chamadas telefônicas via voip e acesso a todo código fonte da aplicação web.

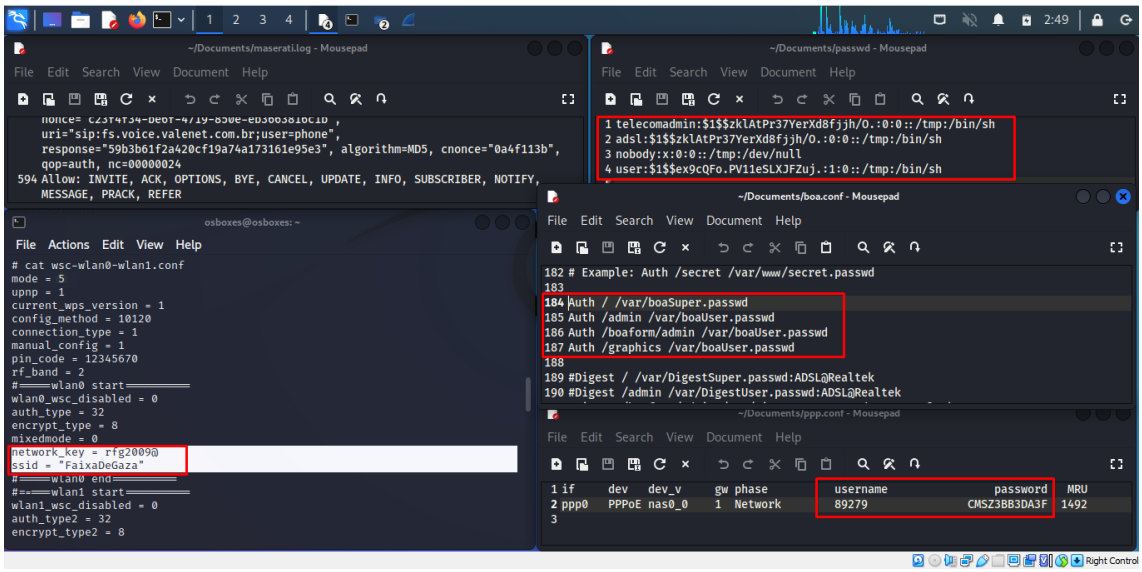

Figura 18 - Arquivos com dados sensíveis encontrados no roteador

Para o ataque de força bruta HTTP(Porta 80), não usaremos o Hydra pois existe outra variável “*PostSecurityFlag*” mostrada na Figura 19, que é passada automaticamente via Js ao enviar o formulário, ela não é aleatória segue um padrão que é validado durante o envio. Caso ela esteja fora desse padrão é retornado uma mensagem de erro de entrada inválida.

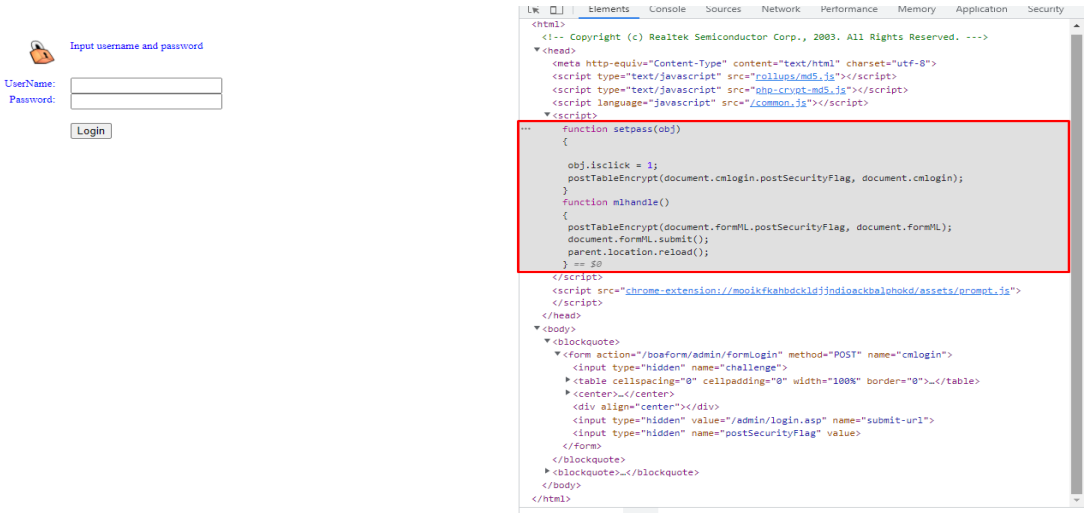

Figura 19 - Segmento do código onde é gerada o valor da variável "PostSecurityFlag"

Vamos partir do pressuposto que não temos acesso aos códigos fonte (conseguido anteriormente via ftp/telnet) para analisar como essa variável é gerada, e iremos construir nossa própria aplicação de força bruta.

Usaremos Python e Sellenium, e para começarmos devemos analisar toda a aplicação mecanismos de segurança e o que ocorre quando a entrada está incorreta. Como podemos observar na Figura 20, ele bloqueia tentativas muito rápidas de login, portanto deve-se implementar um mecanismo de espera no nosso código, isso implica também que será inviável testes com wordlists maiores, portanto deve-se definir muito bem a wordlis't de acordo com o seu alvo.

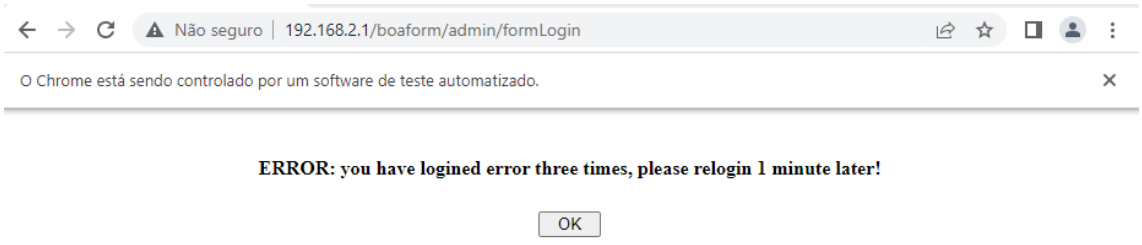

Figura 20 - Erro em tentativas consecutivas de Login

Após inspecionar o código fonte, realizar os testes e executar o script com as mesmas Wordlist's mostradas nas Figuras 13 e 14, temos o resultado mostrado na Figura 21.

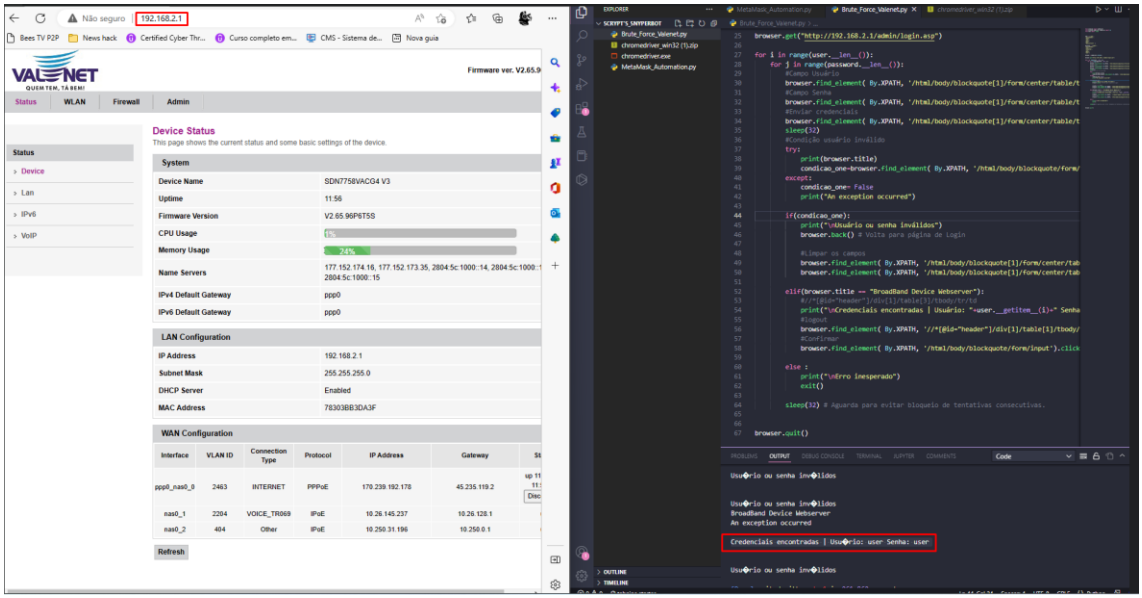

Figura 21 - Execução do script de força Bruta

Foi encontrado um usuário válido, e utilizamos para acessar a interface web. Vale ressaltar que este não é um usuário administrador, pois como podemos observar o arquivo de configuração do serviço web busca o usuário administrador no arquivo /var/boaSuper.passwd mostrado na Figura 22, e neste arquivo o nome do usuário administrador é “telecomadmin”. Porém boa parte desses roteadores de provedor vem com senhas padrões em suas etiquetas e não são alteradas, e em um cenário real o atacante pode usá-las em sua wordlists.

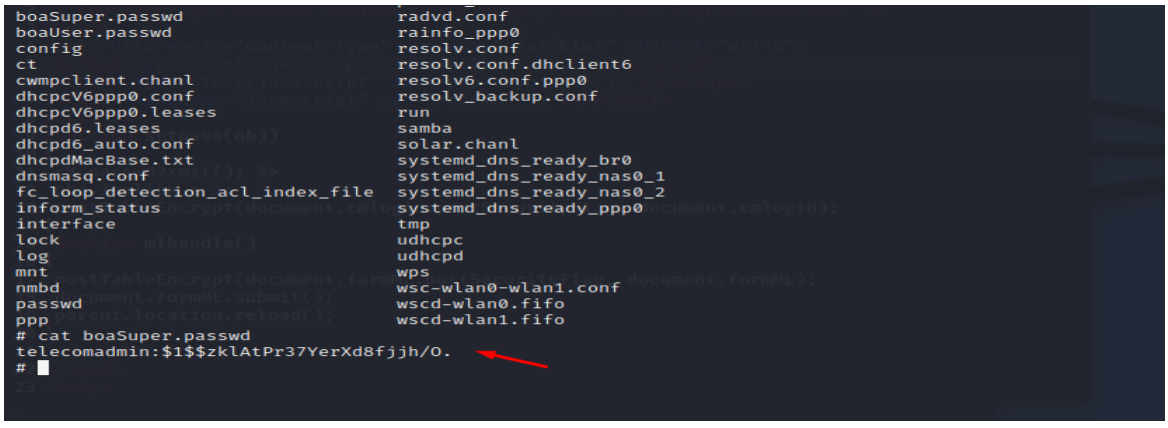

Figura 22 - Credenciais de administração criptografadas com Salt + hash md5

O código fonte do script desenvolvido pode ser encontrado em (Oliv3ira, 2022).

4.2.3 Camêra IP

Figura 23 - Câmera IP

A visualização e gravação das imagens da câmera mostrada na Figura 23, é feita via nuvem, e todas as requisições são feitas através dos seus aplicativos.

Portanto para explorarmos precisaremos analisar os seus aplicativos cujo o link de download pode ser encontrado em Cloud Links. Em nossos testes vamos focar na versão para desktop.

O primeiro ponto que pode ser explorado foi observado logo ao criar uma conta. Após o término do cadastro utilizando um e-mail ou celular a aplicação te gera também um “Id”, e esse Id além de seguir um padrão linear pode ser utilizado para Login na interface da aplicação. O problema aqui é que a própria interface de Login te da dicas caso você digite um usuário válido como pode ser observado na Figura 24, e já sabendo o usuário válido já facilita a vida do atacante pois terá um campo a menos para se preocupar em ataques de força bruta.

Outra opção interessante para ser explorada é essa opção de “Manter senha”, ela foi implementada no intuito de facilitar a vida do usuário na qual não precisará ficar digitando a senha sempre ao logar.

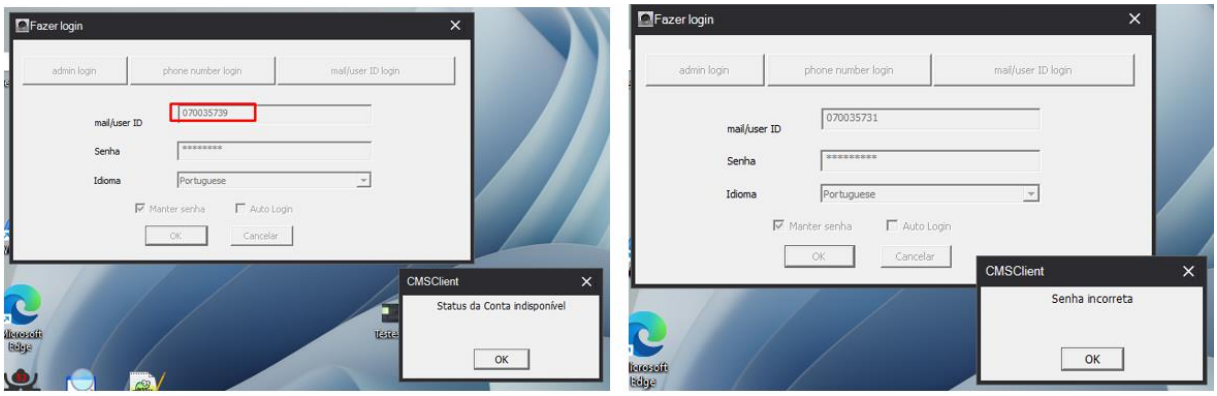

Figura 24 - Diferentes mensagens de erro ao logar, evidenciando usuários válidos

Um formulário é composto de controles, que são seus objetos típicos do Windows, como caixas de texto, rótulos, botões, etc. E cada um desses objetos tem suas próprias propriedades, métodos e eventos. Portanto existem aplicativos que nos permitem visualizer e modificar esses controles um exemplo pode ser visto na imagem logo abaixo.

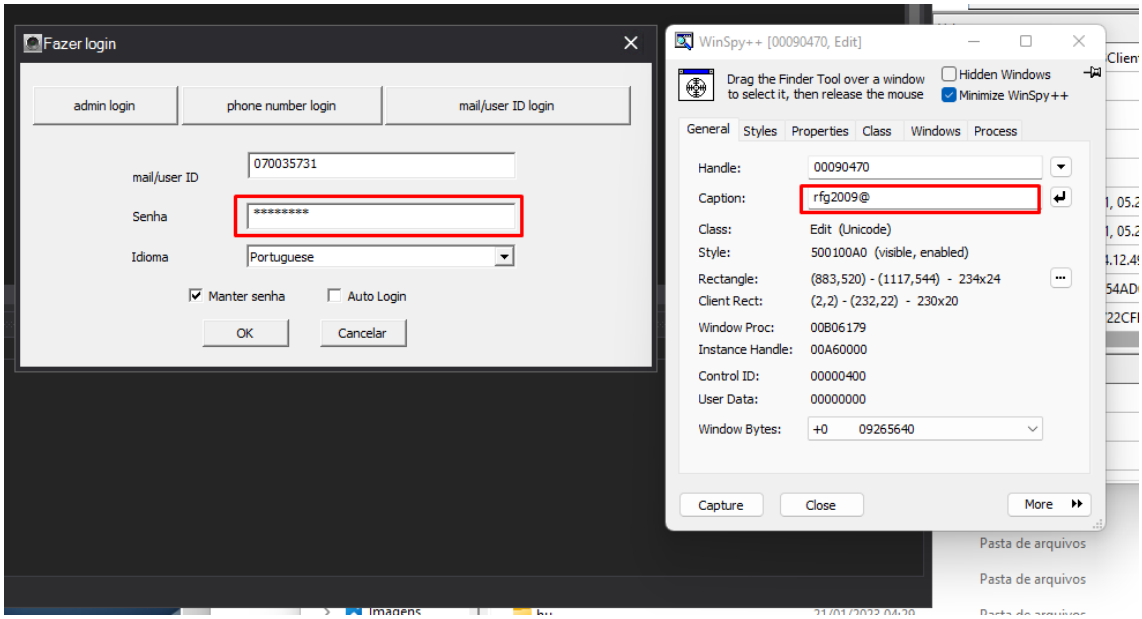

Figura 25 - Utilizando WinSpy++ para descoberta de senhas em formulários

Utilizando o WinSpy++ no nosso formulário de Login podemos identificar que o valor presente no campo “Senha” é armazenado em texto simples como pode ser visto na Figura 25, isso implica que alguém mal intencionado e que tenha ao computador

que está instalada a aplicação pode simplesmente descobrir a sua senha e acessar a câmera em outros dispositivos com as credenciais capturadas.

Existe ainda um método ainda mais simples pra se descobrir a senha, navegando até o diretório da instalação aplicação e dando uma olhada em seus arquivos, podemos encontrar o arquivo com nome de “data.db”, o nome em si já desperta a curiosidade de um atacante. A abrindo este arquivo e decodificando no notepad++ como pode ser visto na Figura 26, podemos inclusive encontrar o “id” e a “senha” do usuário sem qualquer proteção. Obs: Existem outros arquivos com informações valiosas para um atacante, porém seria preciso fazer uma engenharia reversa no APK para entender como eles são usadas, e como podem serem exploradas.

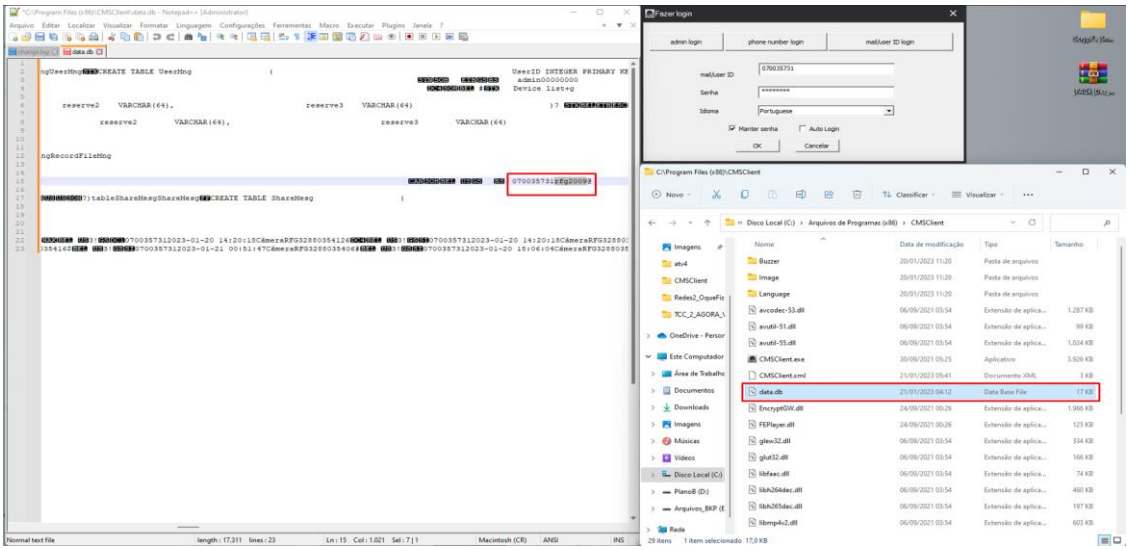

Figura 26 - Arquivos contidos no diretório de instalação da camera

Outra ferramenta interessante para analisarmos essa aplicação é o Wireshark, iremos realizar uma autenticação para entendermos como funciona a comunicação da aplicação. Para fazermos isso, definimos o ip de monitoramento (IP.ADDR==) para o computador que está rodando a aplicação e fazemos login no aplicativo, a captura do pacote pode ser visto na Figura 27.

Assim um atacante consegue entender como a requisição é feita e capturar end-point de comunicação, e neste end-point pode ser realizado uma força bruta no

intuito de detectar mais funções úteis, diretórios ou também pode ser feita umexploração no próprio servidor de comunicação.

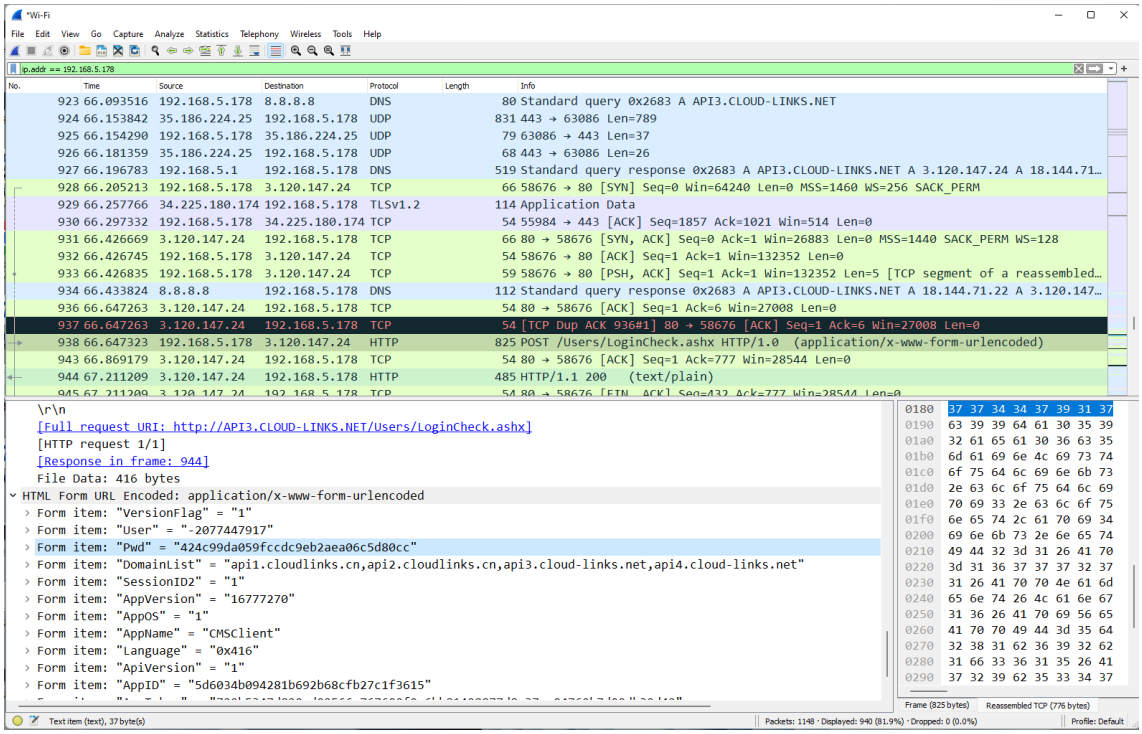

Figura 27 - Wireshark, capturando requisições ao servidor da câmera IP

Na Figura 28, é possível visualizar uma representação exemplificativa de um scan de portas e busca de arquivos e diretórios em uma URL, evidenciando possíveis pontos que poderiam ser explorados por um atacante. Cabe destacar que a câmera em questão realiza gravações em nuvem. Diante disso, surge a seguinte indagação:

"As gravações são efetuadas mesmo em caso de queda do servidor ou de um eventual ataque intencional que resulte na sua queda?".

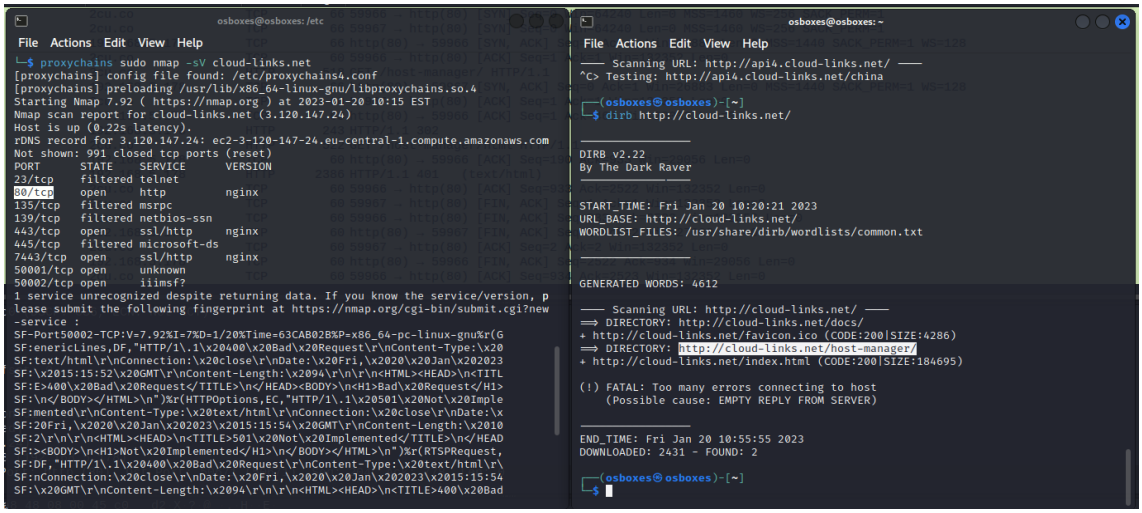

Figura 28 - Fuzing e Varredura de rede no Servidor da Câmera.

A partir deste ponto, não será abordado qualquer detalhamento adicional, visto que tal abordagem poderia violar leis e questões legais, sujeitando-se a possíveis repercussões judiciais. Portanto, apenas foram levantadas anteriormente deduções sobre possíveis pontos de ataque que poderiam ser explorados por indivíduos com intenções maliciosas.

4.2.4 Interruptores Wi-Fi, de marcas distintas.

Figura 29 - Interruptores WiFi (Smart Switch)

Ambos os interruptores Wi-Fi mostrados na Figura 29 compartilham o mesmo aplicativo para comunicação via internet, chamado Tuya SmartLife. Este é um aplicativo baseado na web com uma interface intuitiva para adicionar e controlar dispositivos. Ele possui quatro funções básicas: ligar/desligar, programação, temporização e monitoramento de consumo de energia. Eles se comunicam com o aplicativo usando Wi-Fi 802.11 como método sem fio de conexão.

Antes de examinar o aplicativo em si, vamos realizar testes na rede Wi-Fi. Faremos um ataque de negação de serviço (DoS) que visa interromper a comunicação entre um usuário e um ponto de acesso Wi-Fi. Para realizar este ataque, usaremos ferramentas como iwconfig, airodump-ng, airmon-ng, aireplay-ng e uma interface de rede disponível para atuar no modo monitor.

O primeiro passo é escolher a sua interface de rede e desabilitá-la “ifconfig <interface> down/up” para que fique disponível. Depois, altere-a para o modo monitor usando o comando "iwconfig <sua interface> mode monitor". Em seguida, inicie o monitoramento com o comando "airmon-ng start <sua interface>". Agora você pode

monitorar todo o tráfego Wi-Fi usando o comando "airodump-ng <sua interface>". Isto mostrará informações como o BSSID (endereço MAC do roteador) e o Station (endereços MAC dos clientes conectados ao ponto de acesso). Esses são dados sobre conexões Wi-Fi que estão sendo transmitidas e dentro do alcance da sua interface de rede.

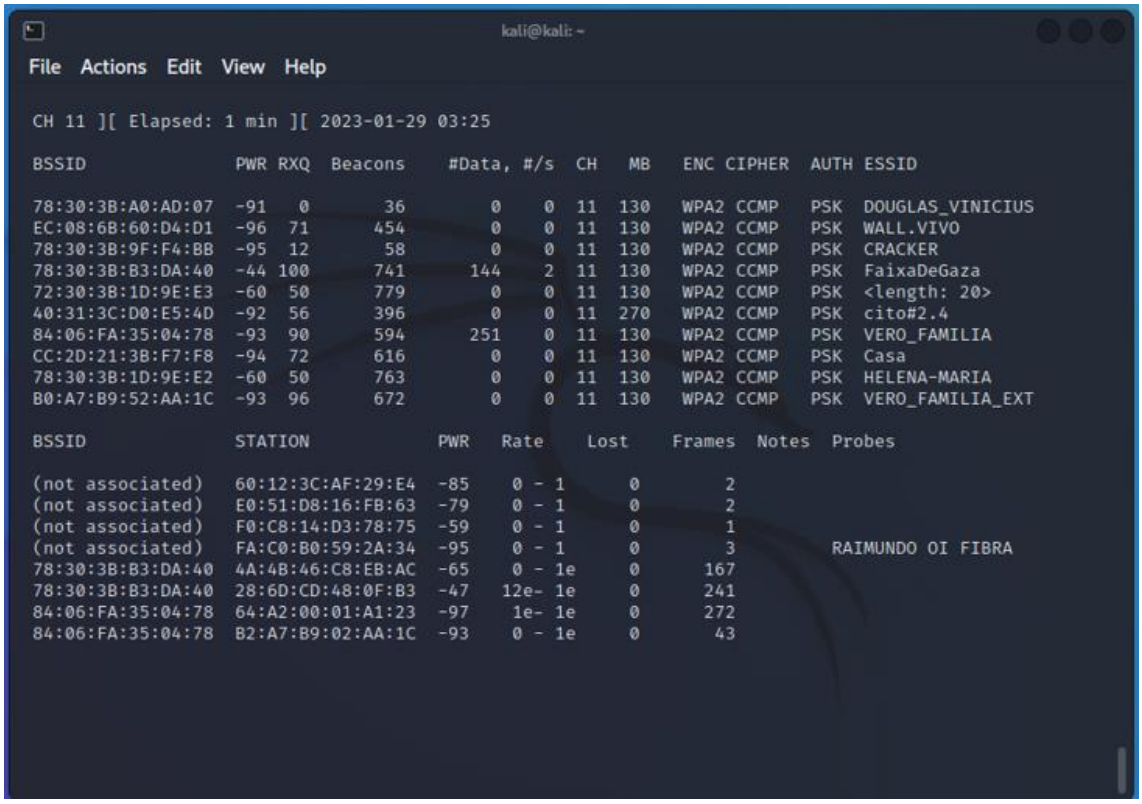

Figura 30 - Escaneamento do Airodump

Como você pode ver na Figura 30, o ESSID já mostra o nome da rede Wi-Fi, então precisamos encontrar o dispositivo alvo do ataque DoS. Se você não souber qual é o endereço MAC do dispositivo que deseja atacar, pode usar ferramentas de ajuda, pois o endereço MAC tem um padrão específico dependendo do fabricante. Um exemplo de ferramenta é o site MacVendors, que possui um banco de dados com vários endereços MAC, a consulta pode ser vista na Figura 31.

Figura 31 - Consulta ao banco de dados do Mac Vendors

Depois de seguir todos os passos acima, basta usar o comando aireplay-ng -0 (opção de desautenticação) <número de pacotes> -a <endereço MAC do roteador> -c <endereço MAC do cliente> <interface>. Durante o ataque, ambos os dispositivos perdem completamente a comunicação com o aplicativo. Isso significa que, se você programou alguma rotina (por exemplo, ligar um dispositivo em uma determinada hora), existe a chance de alguém mal-intencionado impedir que a rotina seja executada. Este ataque não precisa de informações de login ou estar conectado à rede.

Uma maneira de investigar um aplicativo móvel é usando técnicas de engenharia reversa. Isso envolve examinar o código fonte do aplicativo, com a ajuda de ferramentas como o Jadx Decompiler. Ao analisar o código, é possível descobrir informações valiosas, como o banco de dados usado, os métodos de criptografia para armazenar senhas e outras informações sensíveis como pode se observado na Figura 32.

Foi identificado um ponto vulnerável no aplicativo Smart Life IoT, que permite a pareamento do dispositivo por outro celular na mesma rede mesmo o dispositivo já estando pareado em outro celular. Isso resulta na perda de controle do usuário original sobre o dispositivo, e a falta de notificações pode levar a um atraso na detecção do problema.

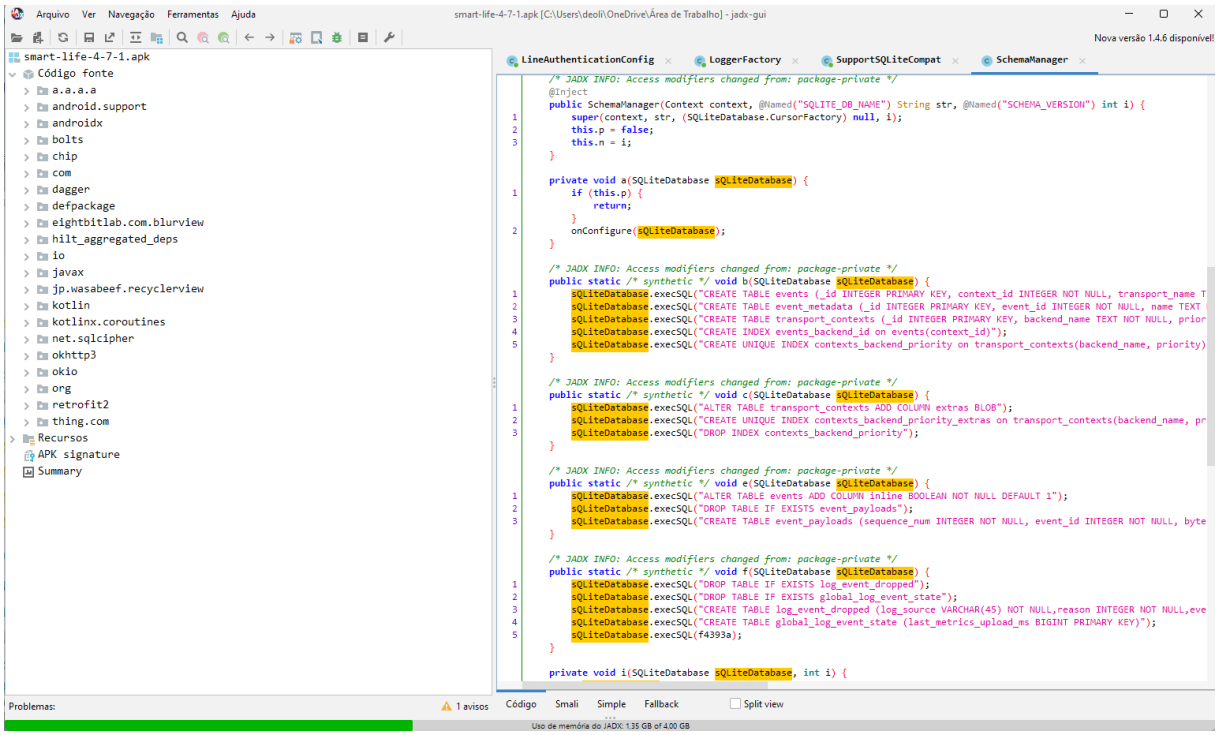

Figura 32 - Código fonte do aplicativo SmartLife

4.2.5 Impressora

Figura 33 - Hp Deskjet Ink advantage 2676

O próximo experimento será com a impressora Hp mostrada na figura 33. Primeiramente, fizemos uma varredura nas portas da impressora mostrado na Figura 34 para entender suas formas de comunicação. Identificamos a porta 9100 aberta, que é utilizada para conexão entre computadores e impressora. A ferramenta PRET é projetada para explorar essa porta, permitindo a comunicação com a impressora usando sua linguagem. De acordo com o repositório oficial da PRET no GitHub, um invasor pode realizar ações como capturar ou alterar trabalhos de impressão, acessar o sistema de arquivos da impressora, armazenar documentos temporariamente, acessar a memória e até mesmo danificar a impressora.

Figura 34 - Scan de rede na impressora utilizando o Nmap

Para explorar uma impressora, é necessário identificar a linguagem que ela usa. Existem três principais linguagens de impressora: PS (Postscript), PJL (Printer Job Language) e PCL (Printer Command Language). A ferramenta PRET é projetada para explorar a comunicação entre os computadores e a impressora usando a linguagem da impressora. Ao passar o endereço IP da impressora como argumento, o PRET pode descobrir a linguagem exata da impressora. Além disso, o PRET pode ser usado para capturar ou manipular trabalhos de impressão, acessar o sistema de arquivos da impressora, armazenar documentos em cache, acessar a memória e até mesmo causar danos na impressora. O modo opcional de debug pode ser usado para resolver problemas. Assim que a tecla enter é pressionada, o PRET oferece acesso à interface de comunicação com a impressora como pode ser observado na figura 35.

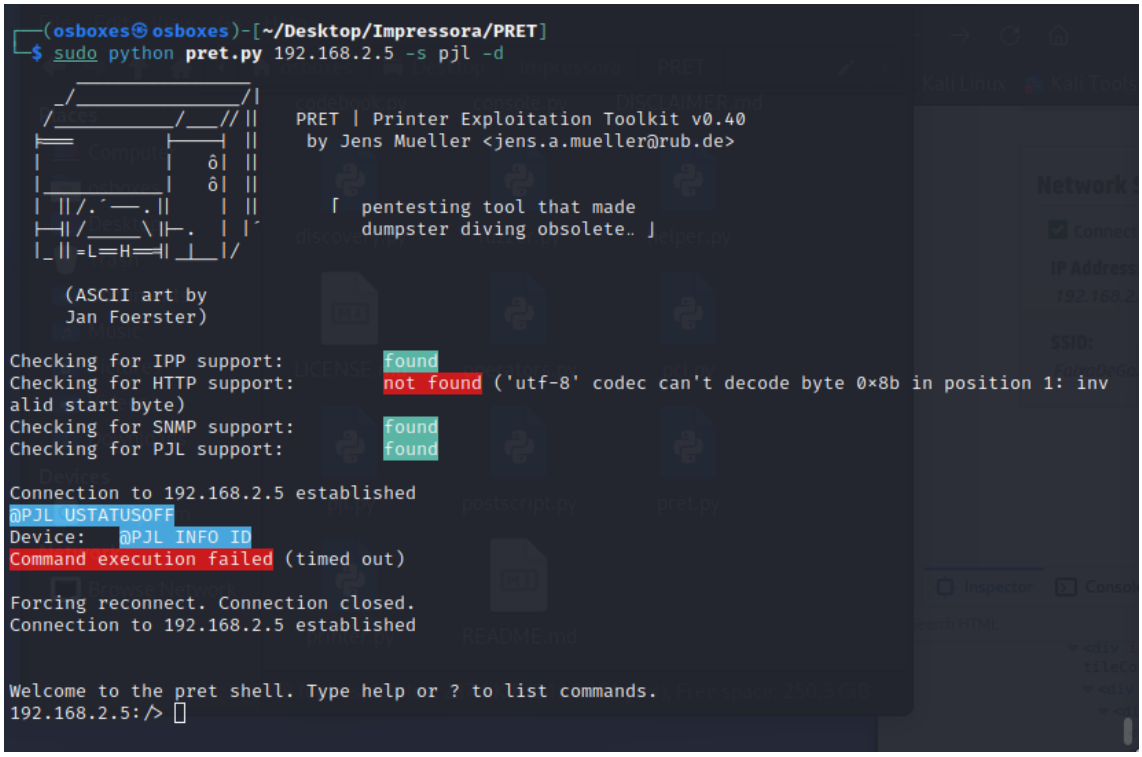

Figura 35 - Execução do PRET

No entanto, ao tentarmos usar os comandos, encontramos um problema. A conexão é interrompida e não nos fornece nenhuma resposta. Verificamos se a solicitação está chegando corretamente na porta de destino usando o Wireshark e descobrimos que, de fato, ela chega, mas algo impede a manutenção da conexão.

Ao pesquisarmos no Issues do repositório observamos que mais pessoas tiveram esse problema, porém ainda sem solução.

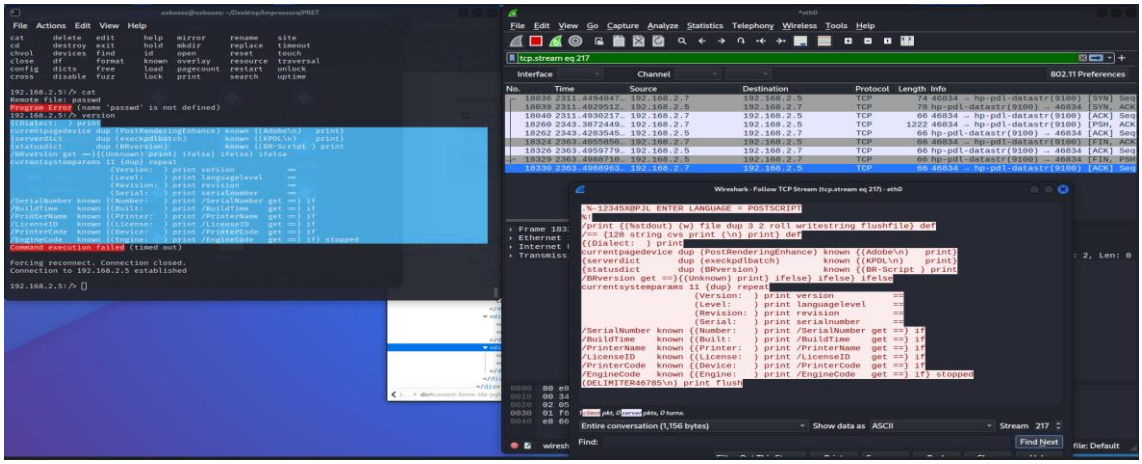

Figura 36 - Análise das requisições do PRET utilizando o Wireshark

Embora tenhamos experimentado vários comandos como mostrado na Figura 36, apenas um funcionou corretamente, que foi "descobrir" mostrado na Figura 37. Ao investigarmos com o Wireshark, descobrimos que este comando usa o protocolo SNMP e, por isso, não apresentou problemas. No entanto, esse comando não fornece informações importantes, pois o nome e o endereço IP da impressora já foram identificados anteriormente através do Nmap.

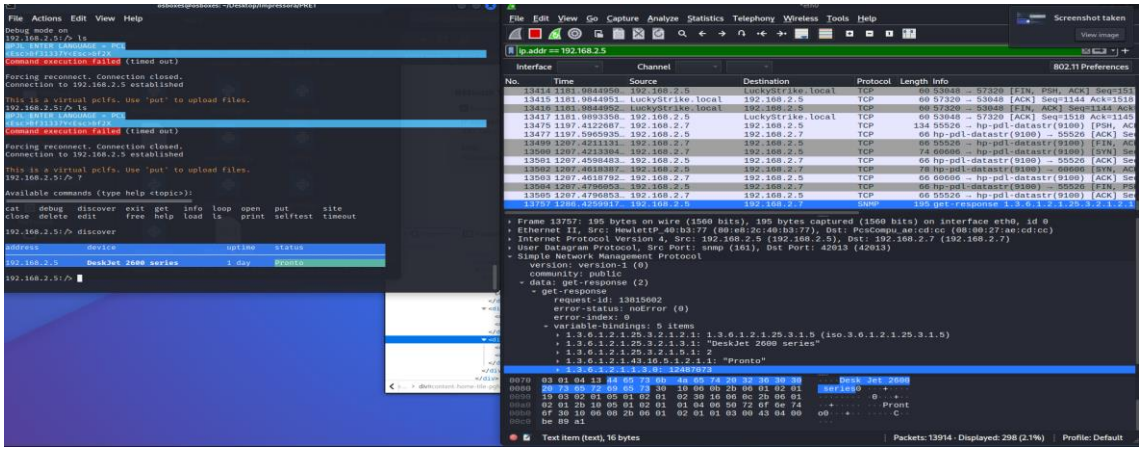

Figura 37 - Analisando o comando "Discover"

Vamos agora entrar na página de administração da impressora que está no servidor web. Para ter uma visão geral rápida das tecnologias usadas no site, estamos utilizando o aplicativo Wappalyzer mostrado na Figura 38. Este aplicativo nos mostra que a tecnologia usada para o *front-end* (parte visual) é o JQuery.

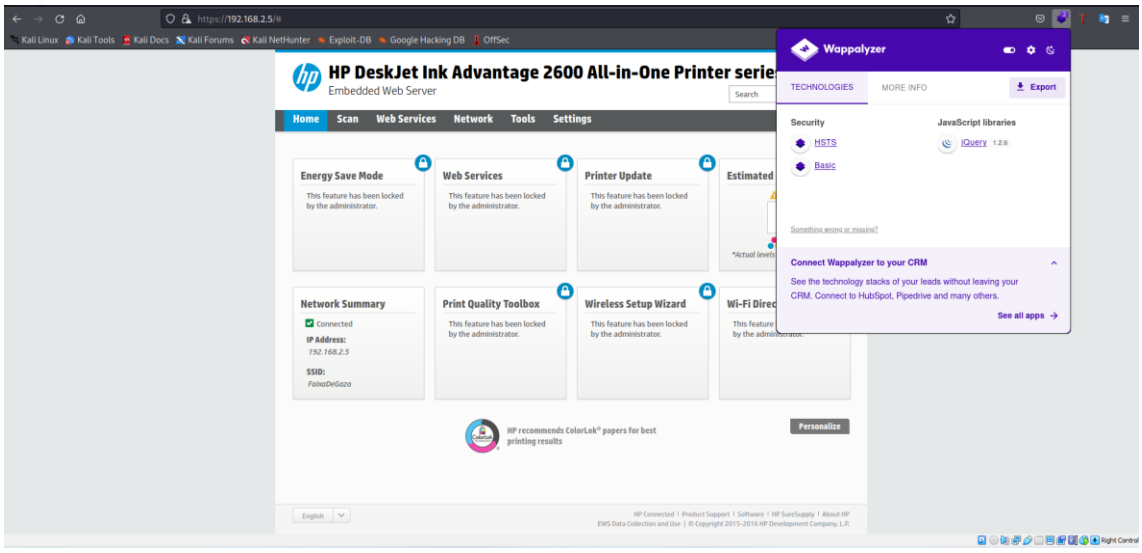

Figura 38 - Serviço que está rodando na porta 80 da impressora

Escreva em outras palavras: Analisamos o código JavaScript e descobrimos vários arquivos, desde configurações até registros de retorno de dados. Infelizmente, poucos desses arquivos exigem autenticação antes de serem visualizados ou usados. Por exemplo, o endereço https://192.168.2.5/Prefetch?type=dtree retorna vários links utilizados na aplicação mostrado na Figura 39, e a maioria deles pode ser acessada sem restrições.

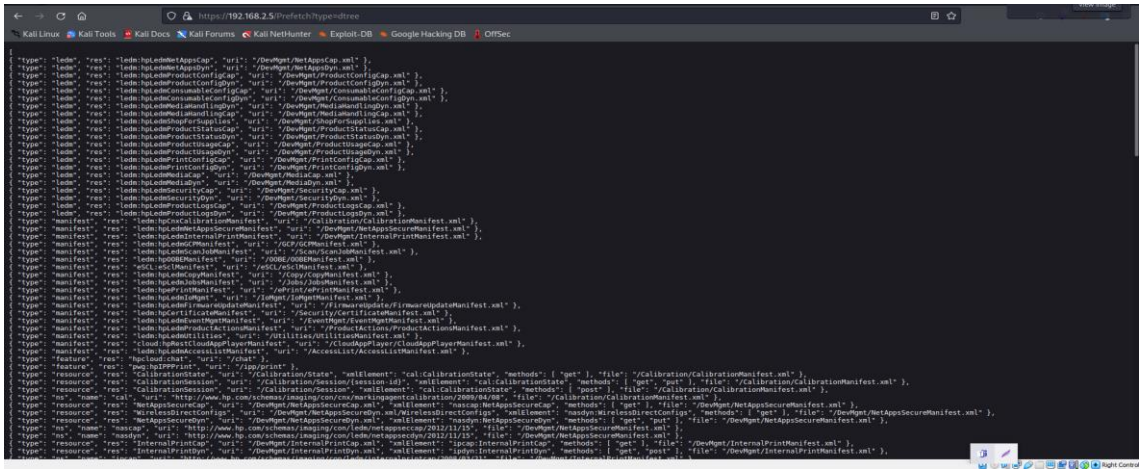

Figura 39 - Retorno da url : https://192.168.2.5/Prefetch?type=dtree

Vamos agora mostrar a capacidade de enviar solicitações de impressão através de duas URLs específicas. A primeira, https://192.168.2.5/Jobs/JobList, retornará o histórico de impressões já realizadas pela impressora. A segunda, https://192.168.2.5/DevMgmt/InternalPrintDyn.xml, permitirá que imprimamos um documento.

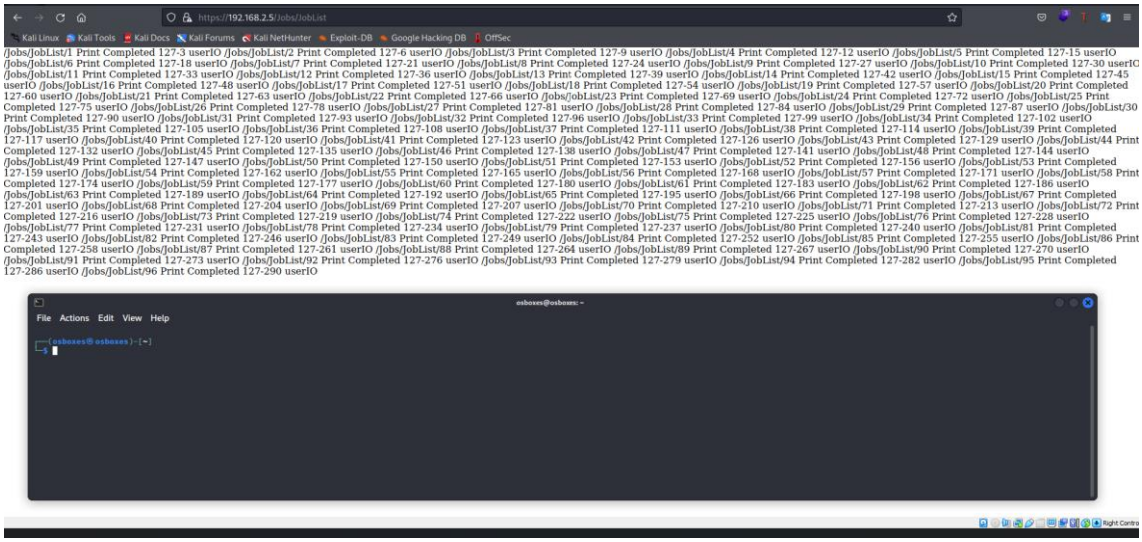

Figura 40 - Retorno da Url : https://192.168.2.5/Jobs/JobList

Vamos verificar se a impressão que vamos realizar irá aparecer na lista de impressões já feitas na impressora, que é retornada pelo arquivo JobList mostrado na Figura 40. Como pode ser observado na Figura 41 se utilizarmos o curl para fazer uma requisição pra url de impressão https://192.168.2.5/DevMgmt/InternalPrintDyn.xml conseguimos imprimir na nossa impressora e também validar a adição dessa impressão no histórico.

Essa exploração faz sentido se essas portas estiverem sendo redirecionas pelos nós anteriores, aí poderíamos sofrer um ataque externo, visando a impressão de spam, sobrecarga na impressora, ou até para os mais criativos a impressão de um qr code que te leva por exemplo pra uma página contendo um malware.

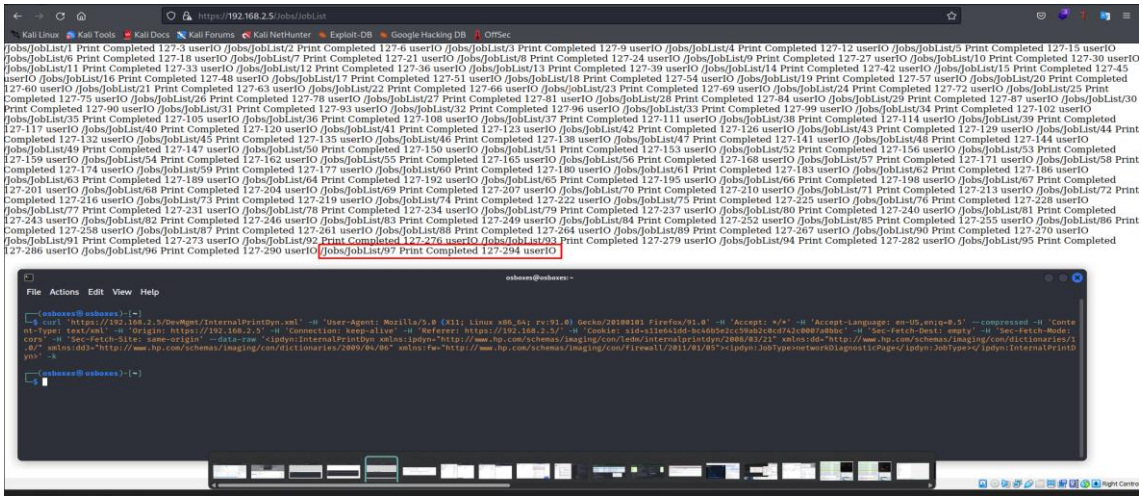

Figura 41 - Realizando a impressão através do Curl

Como mencionado anteriormente, até o momento, para realizarmos os testes, não foi necessário estar autenticado na interface da web. No entanto, há algumas funcionalidades que só estão disponíveis com autenticação, como alterar a senha ou atualizar o firmware.

Neste cenário, as formas de hackear a senha inclui a análise do código-fonte para encontrar uma forma de contornar a função de autenticação ou um ataque de força bruta no formulário de login. No entanto, o site possui medidas de segurança contra ataques de força bruta, então só seria possível realizar ataques com listas de palavras curtas. O formulário pode ser visto na Figura 42.

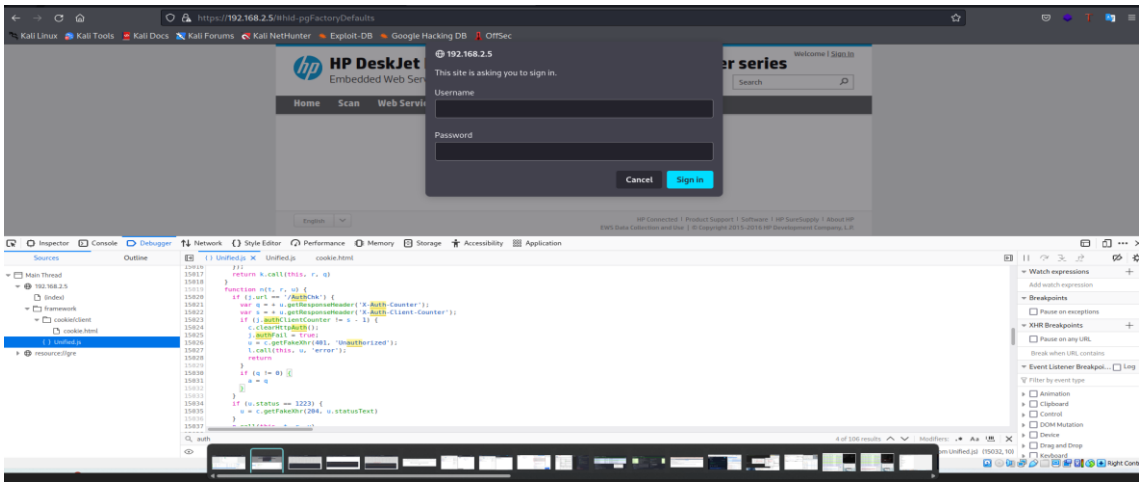

Figura 42 - Formulário de autenticação

4.2.6 Sensor de umidade para solos, desenvolvido com o microcontrolador ESP8266.

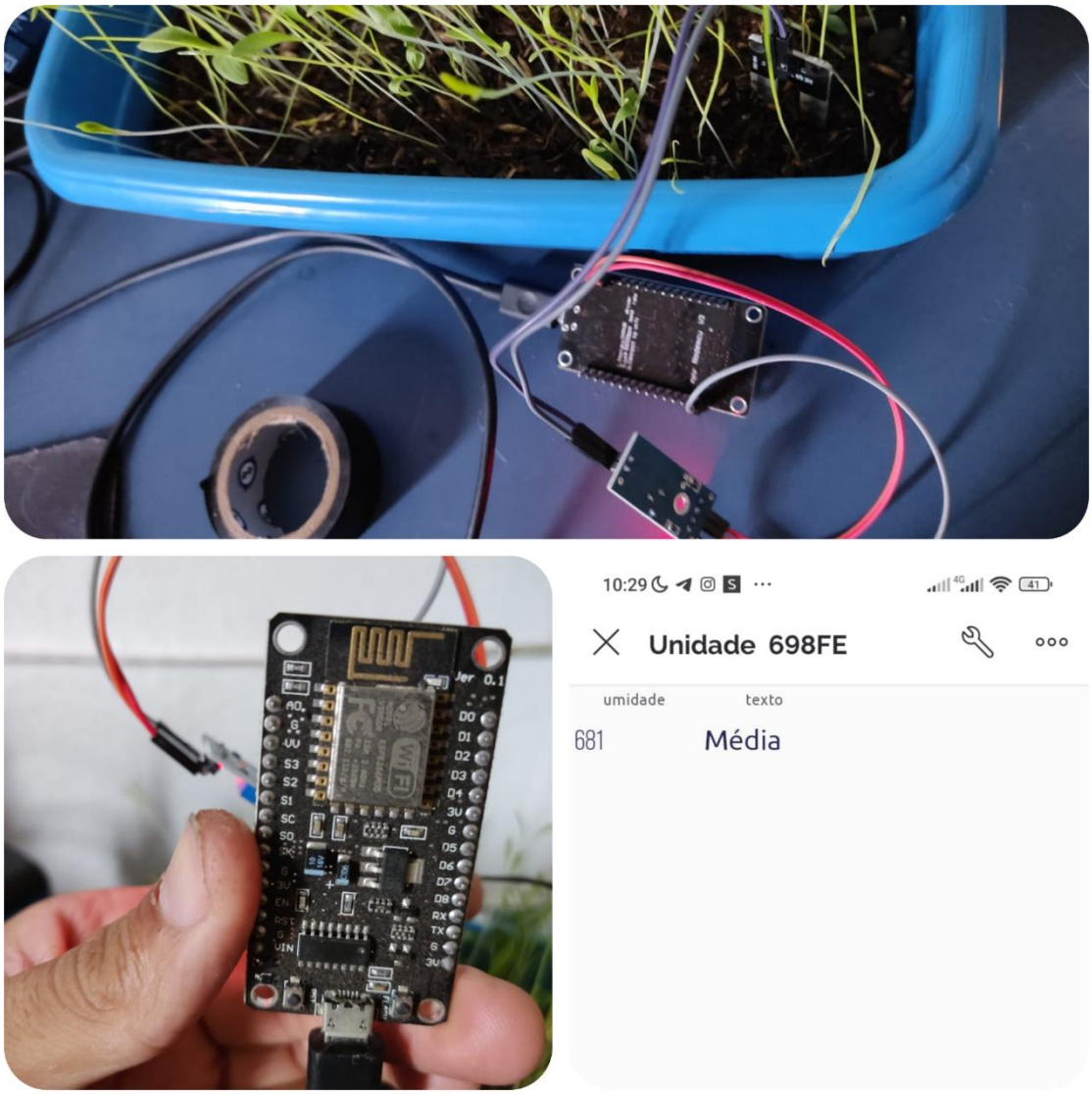

Figura 43 - Placa ESP8266 com sensor de umidade para monitoramento do solo

Esse IoT mostrado na Figura 43, foi desenvolvido seguindo os passos encontrados em (Silva, J. R. ,2023), e foi utilizado um microcontrolador com conectividade Wi-Fi integrada, amplamente utilizada no desenvolvimento de projetos de Internet das Coisas (IoT) e automação residencial, um sensor de umidade e a plataforma Blynk IoT, na qual permite aos usuários criar interfaces de usuário personalizadas para controlar dispositivos conectados à internet. Ele fornece uma biblioteca de widgets e ferramentas de edição de interface do usuário, bem como uma API para conectar dispositivos e trocar dados com serviços de nuvem.

Como podemos o código mostrado no Blog não abre nenhuma porta, ele apenas realiza uma conexão direta com o IoT, portanto um scan de porta para explorar serviços em rede não trará nenhum resultado.

Um hacker pode explorar o ARP Cache Poisoning para capturar o tráfego de rede entre dois nós (Kamar, 2016). Portanto usaremos a ferramenta automatizada de envenenamento de cache para o windows do (Alandau, 2023). Utilizando a ferramenta também conhecida como arpspoof para interceptar o trafego entre microcontrolador e o nosso gateway, podemos observar o IP do servidor que o esp8266 se comunica, porta e protocolos de comunicação. Toda a comunicação com o servidor durante o envio da umidade é criptografada via TSL versão 1.2 que inclusive é a versão mais atual do protocolo, portanto apesar de capturar o tráfego não foi possível ler os dados. Ao iniciar o dispositivo podemos observar que ele também realiza consultas de DNS e DHCP ao roteador e consultas a servidores NTP, a hierarquia dos protocolos, conexões e execução do programa podem ser observadas na figura 44.

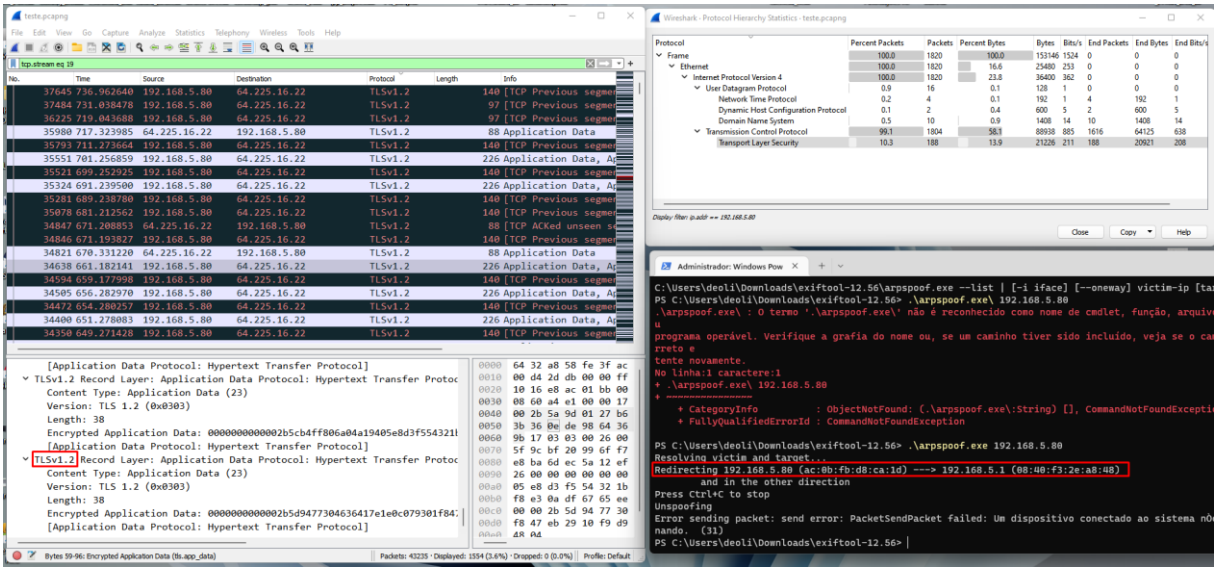

Figura 44 - Ataque de arpspoof e monitoramento através do Wireshark

Para análise do arquivo .apk do aplicativo da Blynk, disponível em https://apkpure.com/blynk-iot/cloud.blynk, utilizaremos desta vez o QARK disponível em https://github.com/linkedin/qark. Além de realizar a engenharia reversa, ele executa algoritmos de análise de código estático para encontrar vulnerabilidades no aplicativo móvel."

A sintaxe para o uso básico é simples “qark –apt <caminho do arquivo .apk >, como pode ser visto na Figura 45, após rodar os comandos ele gera um arquivo html contendo diversas vulnerabilidades que ele conseguiu encontrar.

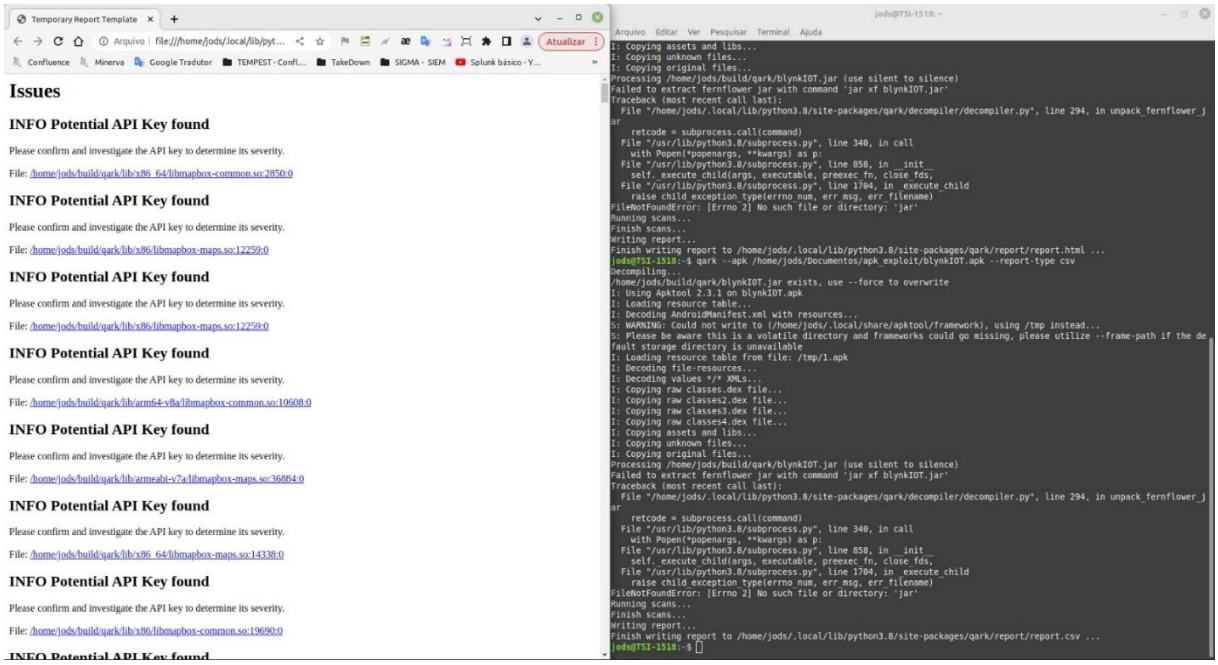

Figura 45 - Arquivo de saída e execução do Código

Apesar de ter encontrado várias, boa parte delas eram falsos positivos como, links de api de fonte, e mapas por exemplo, observe que ele inclusive coloca a informação “Please confirm and Investigate the API key to determine its severity” . Porém ainda sim foi possível encontrar credenciais expostas, nome de tabelas de banco de dados e link’s de alguns serviços.

4.2.7 Google Nest Mini

O Google Nest Mini é um dispositivo inteligente de alto-falante e assistente virtual que funciona com o Google Assistente, ele pode ser visto na Figura 46. Com ele, é possível controlar dispositivos domésticos inteligentes, fazer pesquisas, obter informações sobre o clima e notícias, reproduzir música e muito mais, tudo por meio de comandos de voz ou toques no dispositivo.

Figura 46 - Google Nest Mini

Com intuito de descobrir como o dispositivo troca mensagens, quais são os servidores e se possível ver alguma informação não criptografada, foi realizado um sniffer de redes através do Wireshark em conjunto com o arpspoof (Alandau, 2023) e durante o monitoramento foi efetuado diversos comandos de voz no dispositivo inclusive comandos para ligar e desligar os demais dispositivos inteligentes. Observamos que os comandos estão sendo salvos para a seguinte página https://myactivity.google.com/embedded/product/assistant/, que só pode ser acessado de fato se estiver logado na conta do gmail que a google nest mini está vinculada.

Ao navegar pela página podemos observar na Figura 47 que no console Js, uma mensagem de alerta contra ataques de sequestro de cookies. Os cookies são pequenos arquivos que os sites armazenam no computador do usuário para manter o controle de suas atividades e preferências. Esses arquivos também podem conter informações de login, como nomes de usuário e senhas, e sem eles você precisaria

digitar suas credenciais de acesso toda a vez que fizesse interação com um site que exija autenticação.

Hackers podem usar técnicas de engenharia social ou vulnerabilidades de segurança para sequestrar esses cookies e obter acesso às contas do usuário sem a necessidade de digitar credenciais de login. Portanto iremos replicar esse tipo de ataque no nosso laboratório para observamos o impacto e se de fato é possível realiza-lo.

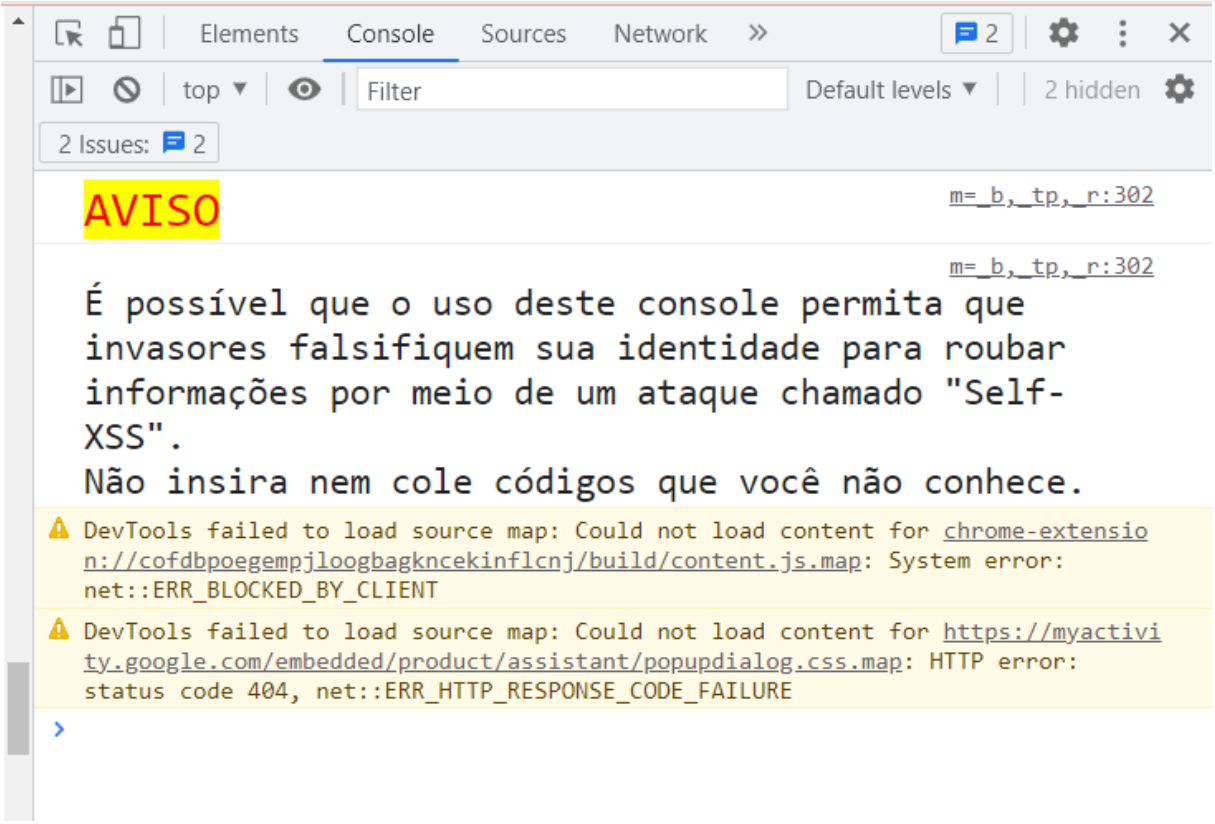

Figura 47 - Mensagem de alerta contra ataques que envolvem sequestro de cookies

Para iniciar esse ataque no nosso laboratório, é necessário estabelecer uma conexão reversa entre a máquina que está sendo atacada e a máquina do atacante. Isso significa que a máquina do atacante precisa estabelecer uma conexão com a máquina alvo e criar uma porta de comunicação para que possa se comunicar com ela. Para criação do nosso payload, utilizaremos a ferramenta Villain, a qual segue a mesma ideia do HoaxShell e tem a proposta de gerar cargas úteis indetectáveis em sistemas operacionais atualizados e com sistemas de segurança (T3l3machus, 2023).

A máquina alvo que utilizaremos roda o Windows 11 e possui o antivírus BitDefender Antivirus Free como pode ser observado na Figura 48, já a máquina atacante será uma máquina virtual do Kali Linux que já estamos utilizando.

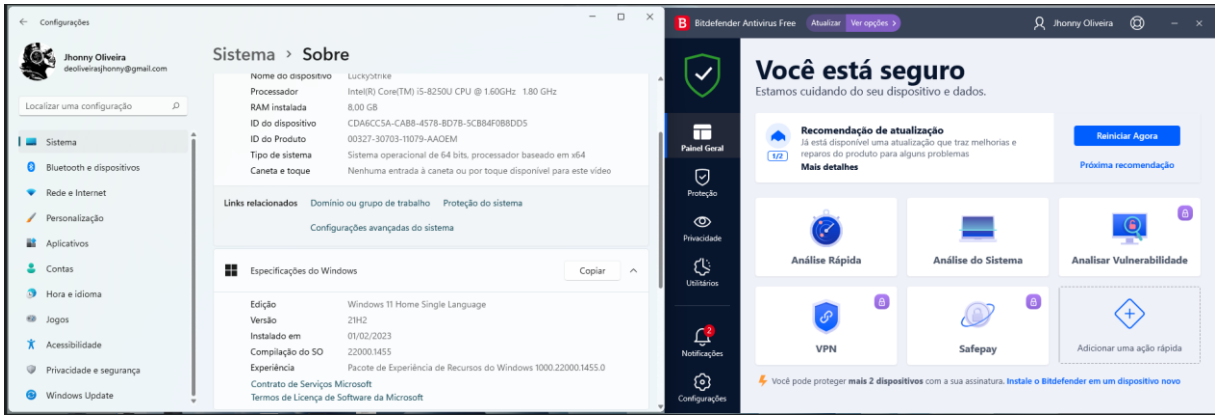

Figura 48 - Configurações da máquina alvo

Durante os testes que fizemos algumas das possibilidades de infecção seria, ou por alterações de dns do roteador que exploraria uma falha, ou impressão do qr code na impressora, ou uma abordagem direta via troca de mensagens em aplicativos, e-mail e etc. Ambos são considerados ataques de engenharia social pois é necessário que a vítima realize alguma ação para se infectar.

Figura 49 - Geração do Payload

Para gerar o *Payload*, como é mostrado na Figura 49, é necessário selecionarmos o sistema operacional que será o alvo do ataque e informar o endereço IP ou a interface de rede que será usada para criar um servidor malicioso que irá ouvir as conexões. Além disso, é preciso definir o parâmetro "*obfuscate*", que utiliza técnicas de ofuscação da carga útil para torná-la indetectável e mais difícil de ser identificada. Dessa forma, o objetivo é tornar o ataque mais efetivo e reduzir a possibilidade de ser detectado pelo sistema de segurança do alvo.

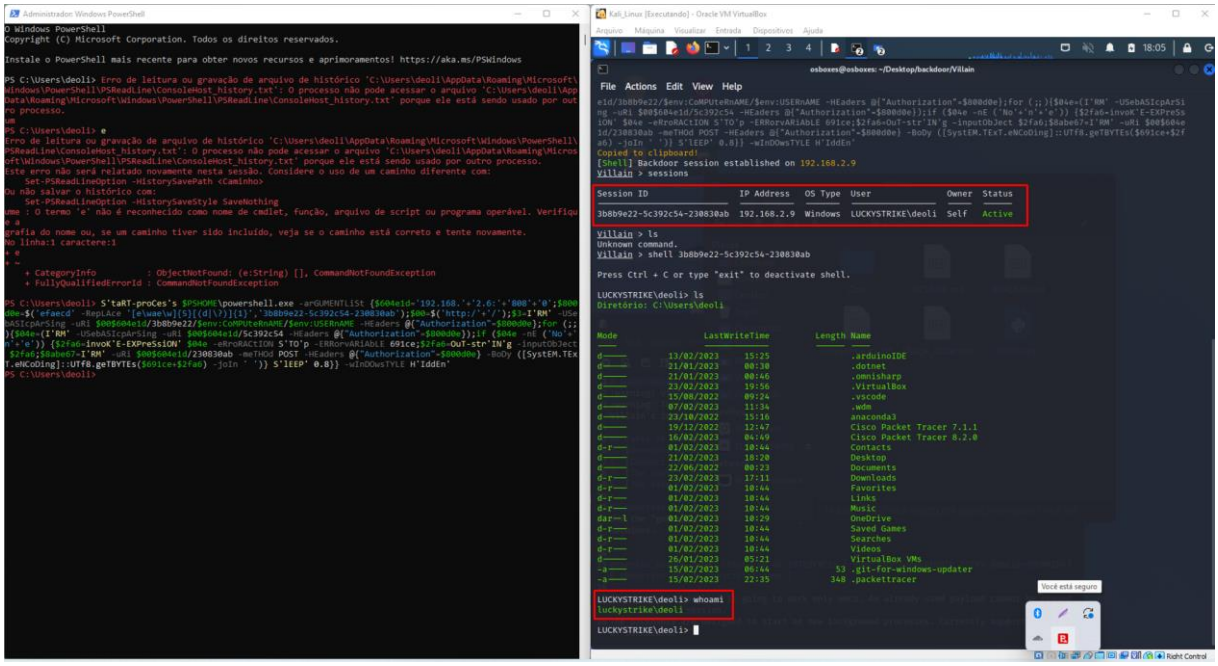

Figura 50 - Execução do payload, e ganho de acesso a shell da máquina alvo

Na Figura 50, é possível ver à esquerda a execução do *Payload* no *powershell* da máquina alvo. À direita, há uma máquina virtual do Kali com a sessão criada e acesso ao shell do Windows. É notável que o sistema de detecção não foi capaz de impedir a infecção.

Agora iremos encontrar o arquivo de cookies do Chrome pois é o navegador que realizamos o acesso a página de histórico do google, ela fica localizada “C:\Users\”nome do usuário”\AppData\Local\Google\Chrome\User Data\”nome do perfil”\Network\Cookies” . O detalhe aqui é que esse arquivo é criptografado, porém não é tão difícil descriptografa-ló se você tem acesso ao computador, pois a chave da criptografia fica armazenada no mesmo diretório. Podemos encontrar um guia detalhado de como descriptografar os cookies do Google Chrome no (The Python Code., 2023) . Usaremos o código fornecido no site para executar o processo de descriptografia e obter acesso aos cookies.

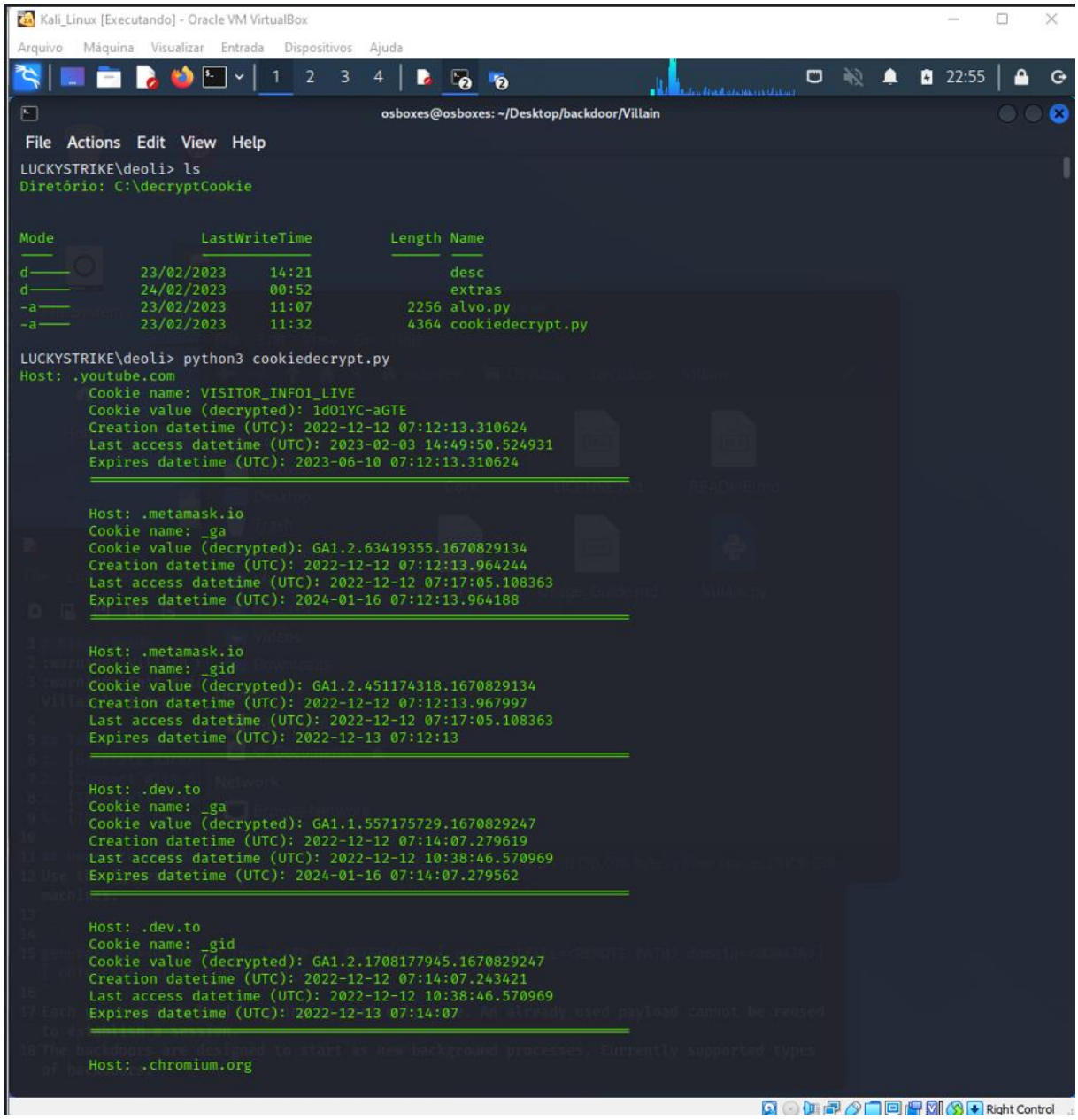

Figura 51 - Descriptografando o arquivo de cookies

Após realizar alguns ajustes no caminho do script, basta executá-lo para obter a exibição dos cookies na tela e um arquivo com os dados descriptografados, como pode ser visto na Figura 51. Agora basta apenas procurarmos os cookies relacionados ao google e configurarmos os cookies no navegador, a configuração do cookie deve ser feita na mesma página na qual ele foi gerado, portanto aqui é necessário logar em outra conta pra termos acesso a página. Para auxiliar na edição dos cookies, é usado uma extensão chamada Cookie Editor, que está disponível no Mozilla Firefox. Com essa extensão, é possível visualizar, editar e excluir cookies de forma fácil e prática.

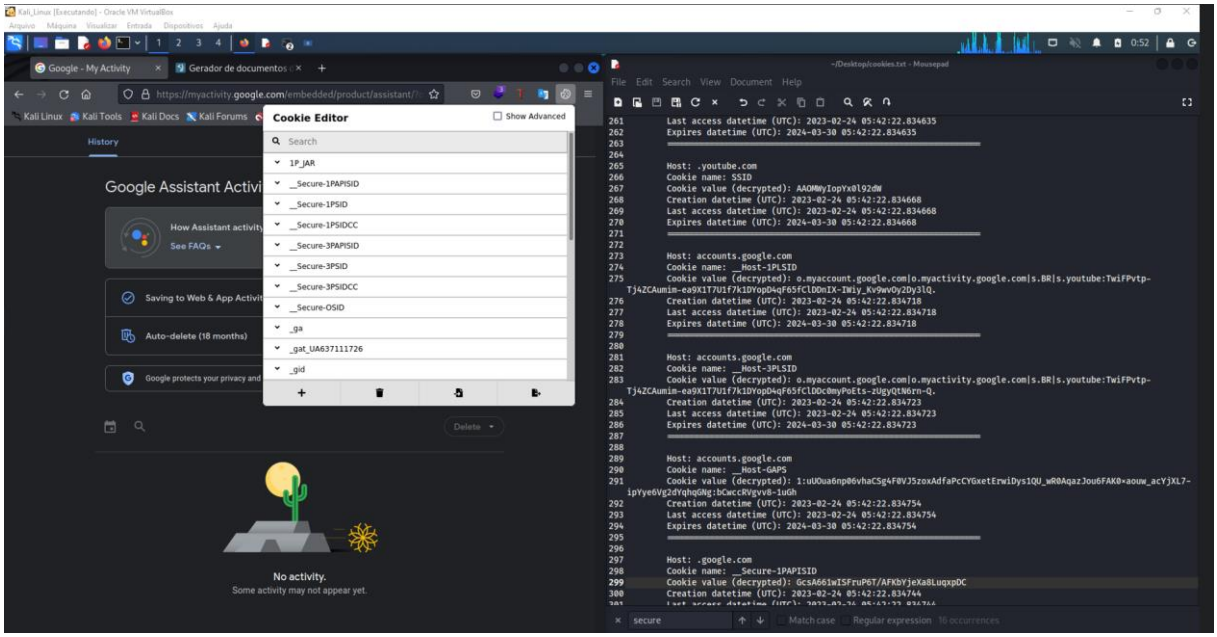

Figura 52 - Alterações dos cookies utilizando a extensão cockie editor

Depois de localizarmos e substituirmos nossos próprios cookies pelos cookies da vítima como é mostrado na Figura 53, obtemos acesso à página da vítima e podemos visualizar o histórico de interações com o Google Nest Mini.

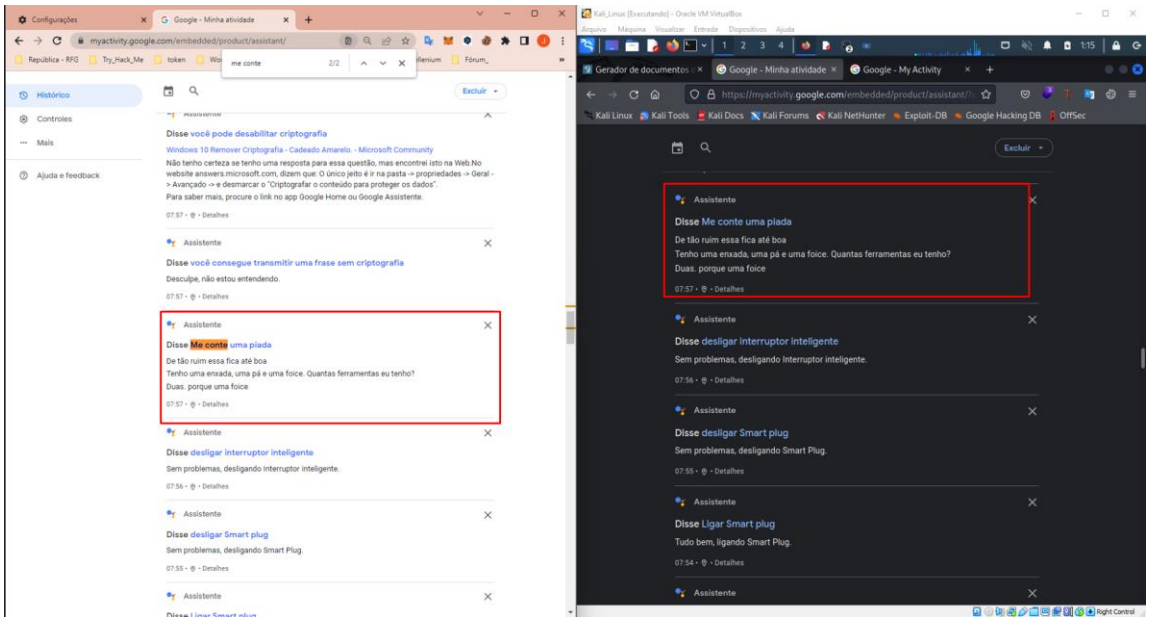

Figura 53 - Sequestro de cookies

Dessa forma, conseguimos acessar informações importantes e confidenciais da vítima pois tem pessoas que utilizam o dispositivo pra agendar reuniões, lembretes ,

encontrar locais para lazer, e etc. Embora não seja uma vulnerabilidade específica da aplicação ou dispositivo, essa técnica representa uma forma adicional de coleta de informações.

4.2.8 Smart TV

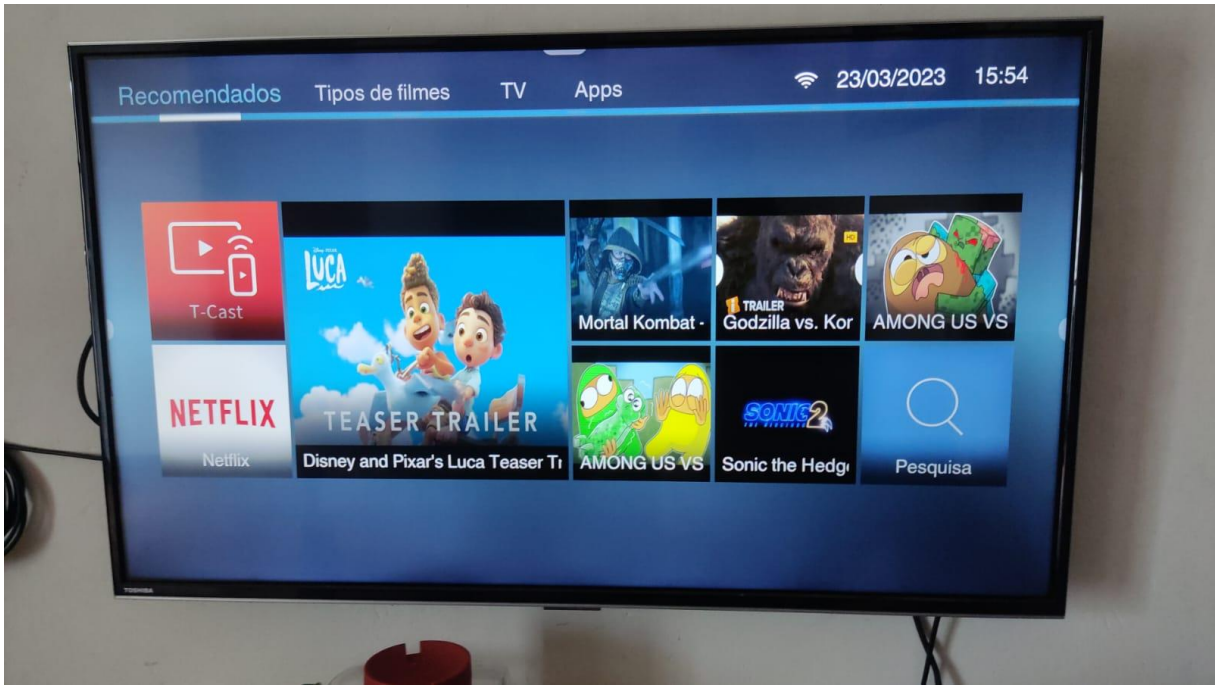

Figura 54 - Smart TV Toshiba

No próximo experimento, foi utilizado uma smart TV mostrada na Figura 54. Ao analisar os pacotes de dados trocados entre essa TV e o roteador, foi possível identificar informações relevantes para a fase de reconhecimento. Por exemplo, a URI "http://x.x.x.x:56790/dd.xml" forneceu informações sobre a própria TV, como evidenciado na Figura 55. Após realizar algumas investigações acerca do modelo da televisão e do serviço em execução na porta 56790, constatamos que essa porta é utilizada para a comunicação da TV por meio de rede local (LAN).

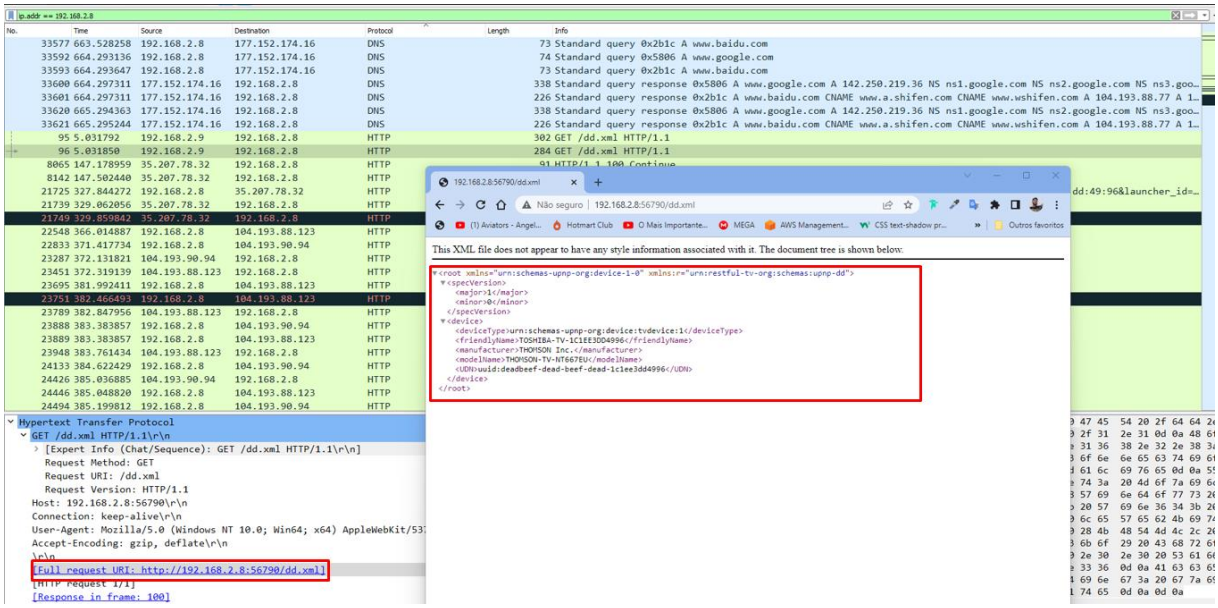

Figura 55 - Sniffing no tráfego da Smart TV

Em relação à essa comunicação, encontramos instruções precisas de como realizar solicitações para a porta em questão em um tópico do fórum (Domoticz, 2023). Contudo, ao tentarmos controlar a televisão com base nessas orientações, não obtivemos sucesso. Embora a URI exista, como pode ser observado na Figura 56, os argumentos passadas retornaram um erro 404 e nada acontece com a televisão.

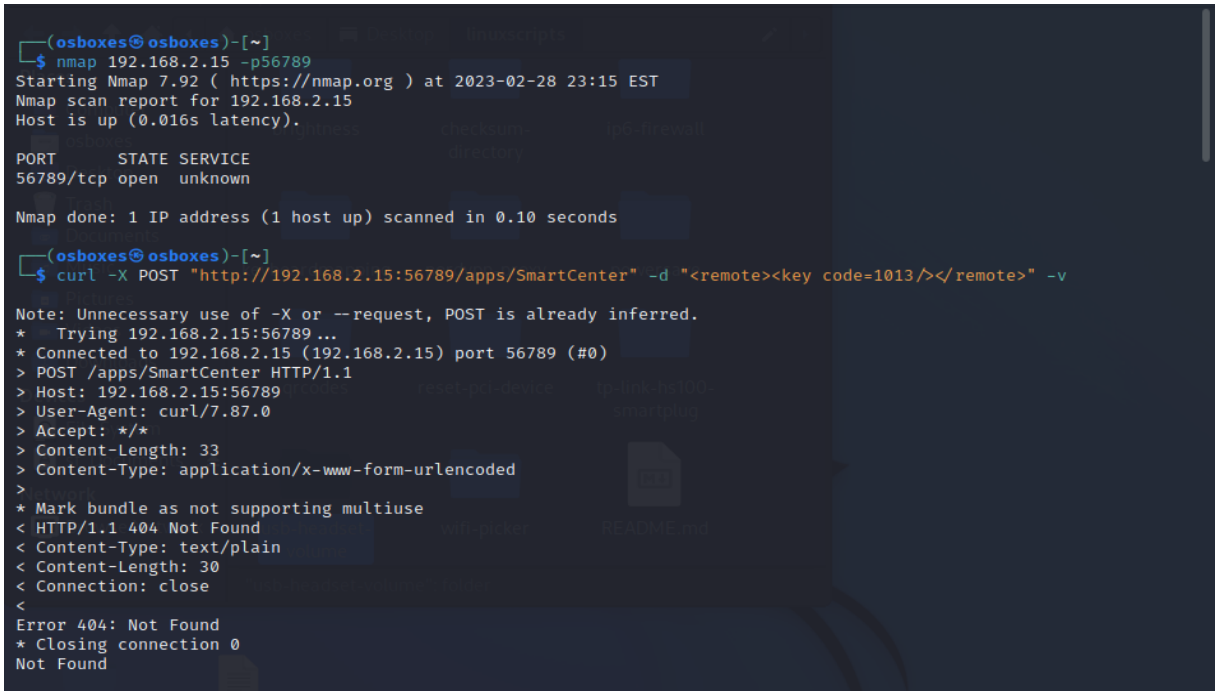

Figura 56 - Utilizando o curl para tentar se comunicar com a api.

No que tange a comunicação o infravermelho, é a comunicação mais conhecida e utilizada, ela está presente em boa parte dos controles remotos. Existem dispositivos que possuem tais sensores o IR Blaster, ele é um dispositivo que emite sinais de infravermelho para controlar aparelhos eletrônicos que normalmente são controlados por controle remoto. Utilizando o IR Blaster conseguimos controlar a televisão, porém eles exigem uma linha de visão direta entre o dispositivo e o receptor de infravermelho, o que significa que eles podem não ser eficazes em ambientes com muita interferência ou obstáculos. O risco aqui é relacionado com a perca de privacidade.

4.2.9 Roteador Mikrotik

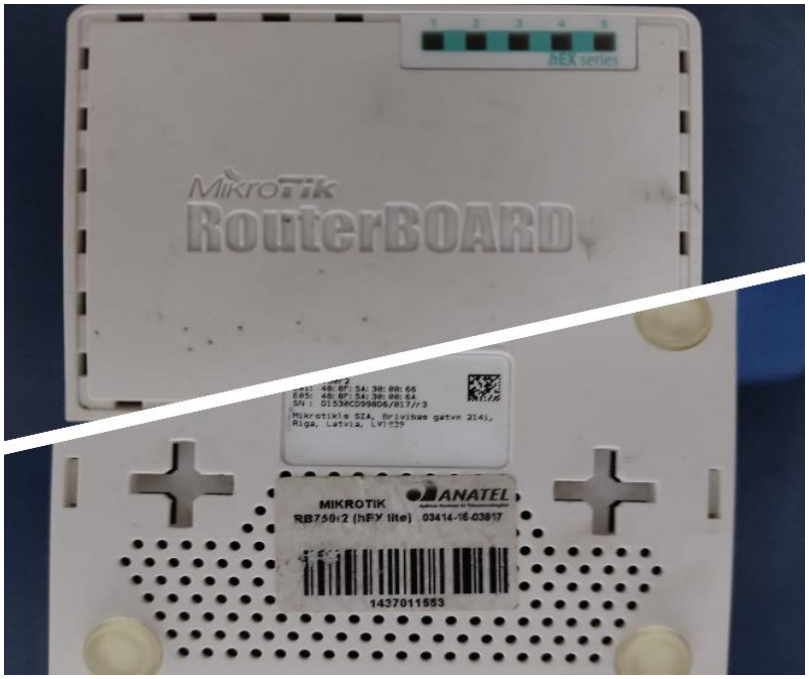

Figura 57 - Mikrotik RB750-R2

O Routerboard RB750-R2 mostrado na Figura 57, é um dispositivo de rede produzido pela empresa MikroTik, amplamente utilizado em pequenas empresas e residências. Ele utiliza o sistema operacional RouterOS, que é conhecido por sua robustez e flexibilidade em gerenciamento de redes. Existem várias formas de acessar o RouterOS, telnet, ssh, web, api e via Winbox cujo a porta padrão é a 8291.

Fizemos algumas investigações com o objetivo de encontrar vulnerabilidades críticas que possam afetar nosso dispositivo e de acordo com a Tenable (s.d.), foi identificada uma vulnerabilidade de vazamento de memória remota no RouterOS Winbox DNS Request. A prova de conceito apresenta um ataque de envenenamento de cache de DNS que pode ser executado sem que o usuário esteja autenticado ele explora a porta 8291 do WinBox.

Envenenamento de cache de DNS (ou DNS cache poisoning, em inglês) é um tipo de ataque que tem como objetivo corromper ou falsificar informações armazenadas no cache de DNS de um servidor ou dispositivo, levando a respostas DNS incorretas. A vulnerabilidade já foi relatada e corrigida nas versões mais recente, porém não é o nosso caso a versão que estamos utilizando no mikrotik é a v6.45.5. stable portanto vamos explora-lá na prática.

Para essa exploração utilizaremos uma terceira máquina virtual nela configuramos um simples servidor dns utilizando dnsmaq e adicionamos algumas entradas com informações falsas de IP/domínio, como pode ser visto na Figura 58.

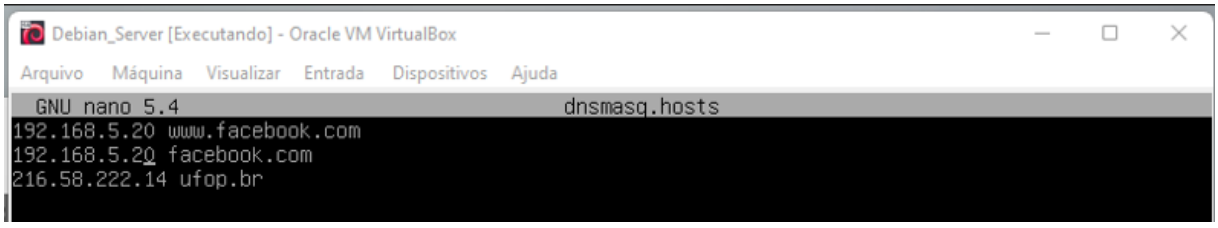

Figura 58 - Arquivo de adição de domínios no dnsmasq

Após realizarmos algumas alterações no script de PoC (Prova de Conceito), escolhendo o domínio ufop.br como alvo e passando os argumentos -i "alvo", -f "servidor de consulta DNS" e -p "porta", foi possível observar que o ataque teve

sucesso.

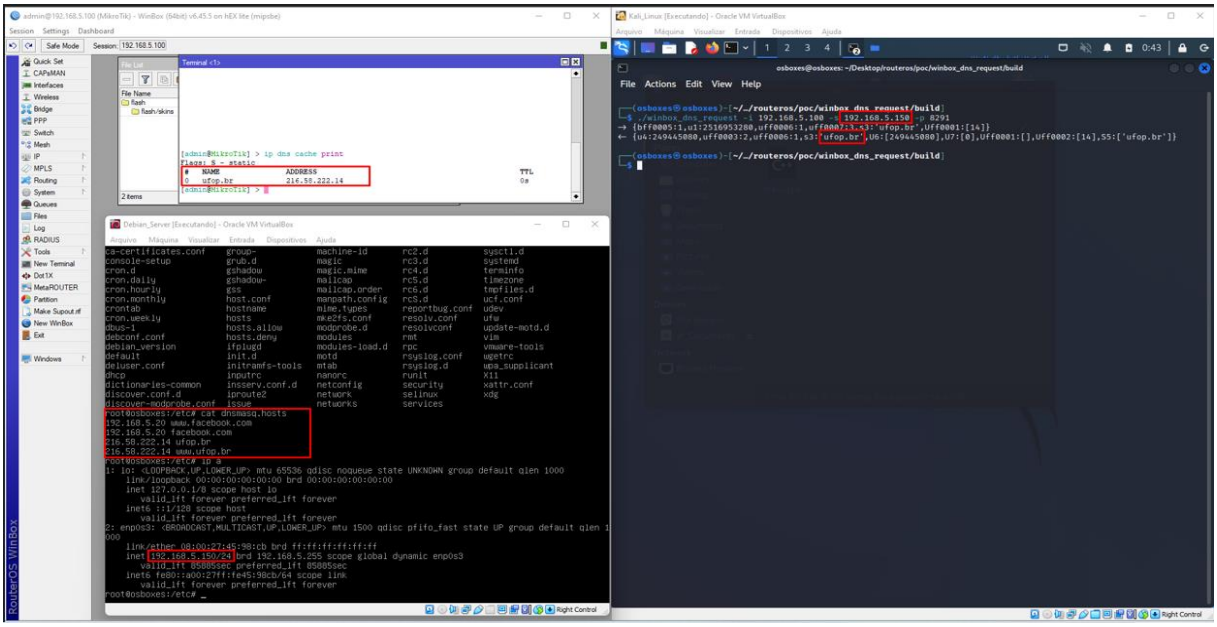

Figura 59 - Envenenamento de cache DNS

Como pode ser observado na Figura 59, ao executarmos o comando "ip dns cache print" no terminal do RouterOS, notamos que o IP associado ao domínio ufop.br não pertencia ao endereço real, indicando que o cache DNS havia sido corrompido e as informações falsas haviam sido armazenadas.

Essa vulnerabilidade representa uma ameaça significativa à segurança do dispositivo, uma vez que pode ser explorada remotamente para redirecionar o tráfego de rede e comprometer a integridade dos dados transmitidos.

4.3 Recomendações de Segurança Proposta

O primeiro passo para garantir a segurança da sua rede residencial é realizar uma verificação das portas abertas e desprotegidas em seu roteador de borda, que é fornecido pelo seu provedor de internet. É importante identificar quais serviços e versões estão sendo executados nessas portas e avaliar se realmente é necessário mantê-las abertas para acesso externo. Além disso, é fundamental avaliar os métodos de acesso a esses serviços, como a autenticação e as credenciais utilizadas, bem como verificar se o roteador possui um firewall que possa impedir o acesso não autorizado. Mesmo que você não tenha um endereço IP válido e esteja dentro da rede

do provedor, outros clientes da mesma rede podem ser capazes de acessar o seu endereço IP e os serviços em execução nele. Uma solução automatizada porém em estágio inicial para verificação de roteadores baseada em firmware é proposto por Toso e Pereira Júnior (2021), a arquitetura proposta por eles inclui um módulo scraper para baixar imagens de firmware de fornecedores de roteadores, um módulo de re-hospedagem para preparar as imagens do firmware para emulação, um módulo de agrupamento para buscar por semelhanças e padrões entre as imagens de firmware, e um módulo de exploração para análise e descoberta de vulnerabilidades nos firmware emulados.

Quando se trata do ambiente interno de rede, é importante garantir a segurança do acesso à rede Wi-Fi e cabeada. Isso pode ser feito usando senhas seguras, além de ter um controle de MAC através de uma lista branca com os dispositivos permitidos, essa funcionalidade está presente em uma ampla gama de roteadores. É também importante criar alertas através de um monitor de redes como por exemplo o Nagios (2019) que possui uma versão gratuita e de código aberto, sobre a presença de dois dispositivos utilizando o mesmo endereço MAC, para evitar acessos não autorizados através de clone de MAC. Porém vale ressaltar que essa solução pode ter efeitos colaterais se implementado em redes maiores, pois muitos desses roteadores apesar de possuírem o controle de MAC não possui um hardware que consiga suprir a alta demanda de solicitações.

É importante verificar todos os dispositivos ativos na rede em busca de serviços abertos e suas versões para identificar vulnerabilidades conhecidas e atualizar ou bloquear o acesso a esses serviços, se necessário. Além disso, é fundamental conscientizar todos os usuários da rede sobre as técnicas de engenharia social usadas pelos cibercriminosos e como se proteger delas. Isso é crucial porque se um dispositivo da rede for infectado, o atacante pode acessar todos os outros dispositivos da rede e até explorar vulnerabilidades não divulgadas nos dispositivos (conhecidas como falhas "zero-day").

5 Conclusões

A partir dos testes realizados e da análise dos resultados obtidos, é possível concluir que os dispositivos testados apresentaram vulnerabilidades significativas em relação ao top 5 da Owasp. Foram encontrados serviços inseguros e senhas frágeis em roteadores de borda e Mikrotik, roteadores mesh com falta de criptografia e informações confidenciais do usuário, além de impressoras e roteadores mesh com interfaces Web inseguras.

Além disso, foi verificado que dispositivos adicionais na rede residencial podem ser usados para obter informações sobre a vítima, como no caso da exploração do serviço que salva informações sobre a Google Nest Mini. Nesses casos, é importante avaliar os riscos de uma exploração bem-sucedida por meio de engenharia social.

Por outro lado, dispositivos como interruptores inteligentes e TVs inteligentes apresentam menos possibilidades de ataque, mas podem levar à falta de privacidade e importunação do usuário. Portanto, é essencial que os usuários estejam cientes das vulnerabilidades em seus dispositivos e tomem medidas para mitigar esses riscos, a fim de garantir a segurança e privacidade de seus dados na rede residencial.

5.1 Limitações e Trabalhos Futuros

Uma limitação potencial do estudo é a abrangência dos testes em relação a outros dispositivos e vulnerabilidades indicadas pela OWASP. Os resultados obtidos podem não ser aplicáveis a todas as possíveis ameaças e vulnerabilidades existentes em outros dispositivos que não foram testados, o que limita a generalização dos achados do estudo.

Algumas sugestões de possíveis trabalhos futuros com base nos resultados e metodologia deste estudo são:

- Ampliar a abrangência dos testes para incluir outros dispositivos e vulnerabilidades indicadas pela OWASP, a fim de verificar as ameaças mais recentes e obter uma compreensão mais abrangente das possíveis vulnerabilidades em diferentes tipos de dispositivos.
- Avaliar a eficácia de medidas de mitigação específicas para as vulnerabilidades identificadas neste estudo, a fim de determinar a eficácia de abordagens de correção e proteção para dispositivos vulneráveis.

- Realizar investigações detalhadas sobre a eficácia de diferentes soluções de segurança de rede residencial, como firewalls, softwares antivírus e programas de detecção de intrusão, a fim de avaliar sua eficácia na prevenção de ataques e proteção dos dados do usuário em diferentes cenários e configurações.
- Avaliar a conscientização e educação dos usuários em relação à segurança cibernética em redes residenciais, com o objetivo de identificar estratégias eficazes de educação e treinamento para promover a adoção de práticas de segurança adequadas.
- Investigar a adoção de melhores práticas de segurança por parte dos fabricantes de dispositivos para minimizar vulnerabilidades em seus produtos e proteger os usuários contra ameaças cibernéticas.

Referências

MAGRANI, Eduardo. A Internet das Coisas. 1. Ed Rio de Janeiro: FGV Editora, 2018.

TENENBAUM, Andrew S. Redes de computadores. 4. Ed Rio de Janeiro:Campus Editora, 2003.

Van Kranenburg, Rob. "The internet of things." World Affairs: The Journal of International Issues 15.4 (2011): 126-141.

IoT Analytics. (2022). Number of Connected IoT Devices. IoT Analytics. Disponível em: https://iot-analytics.com/number-connected-iot-devices/. Acesso em: 11 de Abr. 2023.

Open Web Application Security Project. (2018). OWASP Internet of Things (IoT) Project. Acessado em 11 de abril de 2023, de https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project.

FOLHA DE S.PAULO. Home office é adotado por 33% das empresas no Brasil, diz FGV. Disponível em: https://www1.folha.uol.com.br/mercado/2023/01/home-office-e-adotado-por-33-das-empresas-no-brasil-diz-fgv.shtml. Acesso em: 11 de abril de 2023.

TOSO, Gianluigi Dal; PEREIRA JÚNIOR, Lourenço Alves. Enumeração de sistemas operacionais e serviços de firmwares de roteadores sem-fio. In: WORKSHOP DE TRABALHOS DE INICIAÇÃO CIENTÍFICA E DE GRADUAÇÃO - SIMPÓSIO BRASILEIRO DE SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS (SBSEG), 21., 2021, Evento Online. Anais... Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 178-191. DOI: https://doi.org/10.5753/sbseg_estendido.2021.17351.

LEITE, Leandro Rogério Corrêa. Internet das Coisas (IoT): vulnerabilidades de segurança e desafios. 2019.

BERLANDA, Rodrigo Grando. Guia de segurança da informação para a conectividade de dispositivos IoT. 2021.

FUKUDA, Leonardo Massami. Segurança da informação em IOT. 2019.

MOSENIA, Arsalan; JHA, Niraj K. A comprehensive study of security of internet-of-things. IEEE Transactions on emerging topics in computing, v. 5, n. 4, p. 586-602, 2016.

Oliv3ira. (2022). BruteForce_ONU_HTTP.Service-BroadBand-Device-Webserver [Software]. Acessado em 23 de março em https://github.com/0liv3ira/BruteForce_ONU_HTTP.Service-BroadBand-Device-Webserver.git

Cloud-Links.net. (s.d.). Cloud-Links.net [Website]. Acessado em 23 de março de 2023 de http://cloud-links.net/index_en.html

MAC Vendors. MAC Address Lookup [Website]. Acessado em 23 de março de 2023 de https://macvendors.com/

Tuya. (s.d.). All-in-One App [Aplicativo móvel]. Acessado em 23 de março de 2023 de https://www.tuya.com/product/app-management/all-in-one-app

RUB-NDS. (s.d.). PRET [Repositório de software]. GitHub. Acessado em 23 de março de 2023 de https://github.com/RUB-NDS/PRET

RUB-NDS. (2018, 18 de janeiro). Don't work with https protocol (Issue #68) [Questão de software]. GitHub. https://github.com/RUB-NDS/PRET/issues/68

Silva, J. R. (2023). Monitoramento de umidade do solo com uso de Blynk IoT e ESP8266. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) - Universidade Federal de Minas Gerais, Belo Horizonte, MG.

Blynk. (2023). Blynk. Acessado em 23 de março de 2023, de https://blynk.io/

Alandau. (2023). Arpspoof. Acessado em 23 de março de 2023, de https://github.com/alandau/arpspoof.git

QARK. (2023). Linkedin QARK. Acessado em 23 de março de 2023, de https://github.com/linkedin/qark

APKPure. (2023). Blynk IoT. Acessado em 23 de março de 2023, de https://apkpure.com/blynk-iot/cloud.blynk

Google. (2023). Google - My Activity. Acessado em 23 de março de 2023, de https://myactivity.google.com/embedded/product/assistant/

T3l3machus. (2023). Villain. Acessado em 23 de março de 2023, de https://github.com/t3l3machus/Villain.

The Python Code. (2023). Extract Chrome Cookies Python. Acessado em 23 de março de 2023, de https://www.thepythoncode.com/article/extract-chrome-cookies-python.

Mozilla. (2023). Cookie Editor. Acessado em 23 de março de 2023, de https://addons.mozilla.org/en-US/firefox/addon/cookie-editor/

Domoticz. (2023). How to install Domoticz on a Raspberry Pi. Acessado em 23 de março de 2023, de https://www.domoticz.com/forum/viewtopic.php?t=26042

Tenable. (s.d.). RouterOS Winbox DNS Request Remote Memory Leak Vulnerability. Acessado em 23 de março de 2023, de https://github.com/tenable/routeros/tree/master/poc/winbox_dns_request.

Loi, F., Sivanathan, A., Gharakheili, HH, Radford, A. e Sivaraman, V.(2017). Avaliando sistematicamente a segurança e a privacidade dos dispositivos IoT do consumidor.

Em Proc. do 2017 Workshop on IoT Security and Pri vacy - IoTS&P '17, Dallas, Texas, EUA. ACM Nova York.

Pesce, L. (2017). Sans webcast: Eu não dou a mínima – apresentando a metodologia de ataque iot – youtube.

Kumar, S., & Reddy, A. R. (2016). A Survey on ARP Spoofing: Threats, Prevention and Detection Techniques. International Journal of Computer Applications, 134(1), 13-20.

Hashes.com. Hash Identifier. Acessado em 26 de março de 2023, de: https://hashes.com/en/tools/hash_identifier.