Congresso de Segurança da Informação das Fatec

Estudo Prático de Ataques *Juice Jacking* em Dispositivos Android

Practical Study of Juice Jacking Attacks on Android Devices

Gustavo Genoni Gonçalves, Faculdade de Tecnologia de Ourinhos,
gustavo.goncalves25@fatec.sp.gov.br

Beatriz Juliana Oliveira, Faculdade de Tecnologia de Ourinhos,
beatriz.oliveira64@fatec.sp.gov.br

Paulo Roberto Galego Hernandes Junior, Faculdade de Tecnologia
de Ourinhos, paulo.galego@fatecourinhos.edu.br

Resumo

Neste trabalho, almeja-se apresentar de maneira prática a ocorrência e o funcionamento do ataque conhecido como *Juice Jacking*, um tipo de ataque que explora a transferência de dados de dispositivos móveis sem o consentimento do usuário, quando este conecta seu smartphone a uma estação de carregamento comprometida. É importante elucidar tanto indivíduos quanto organizações sobre essa forma de crime, uma vez que, frequentemente, ele ocorre sem acionar alarmes perceptíveis para o usuário, tornando-se, assim, uma ameaça ainda mais perigosa. O intuito deste ataque é furtar informações de dispositivos móveis por meio de estações de carregamento USB.
Palavras-chave: USB, Juice Jacking, Ataque, dados, estação carregamento

Abstract

This paper aims to present in a practical way the occurrence and functioning of the attack known as *Juice Jacking*, a type of attack that exploits the transfer of data from mobile devices without the user's consent, when the user connects his/her smartphone to a compromised charging station. It is important to educate both individuals and organizations about this type of crime, since it often occurs without triggering noticeable alarms for the user, thus becoming an even more dangerous threat. The purpose of this attack is to steal information from mobile devices through USB charging stations.

1. Introdução

Até o momento da presente pesquisa, onde a dependência de dispositivos móveis é inegável, a conveniência de carregar nossos *smartphones* e *tablets* em estações públicas tornou-se uma prática comum, como presente na análise de Schneier (2015). No entanto, à medida que a tecnologia avança, surgem novas ameaças, e uma delas é conhecida como *Juice Jacking*. Este evento refere-se à manipulação maliciosa das estações de carregamento público por cibercriminosos, que buscam extrair informações sensíveis de qualquer pessoa que conecte seu dispositivo Android, via USB, para recarregar. Essa forma de ataque levanta sérias preocupações quanto à segurança dos dados pessoais e corporativos, como em Nguyen (2017) que demonstrou em prática.

Neste contexto, a justificativa para a elaboração deste artigo acadêmico é explorar as nuances do *Juice Jacking*, os riscos associados e as medidas que os usuários podem adotar para proteger suas informações em um cenário cada vez mais vulnerável.

Essa hipótese pressupõe que a simulação de *Juice Jacking* não apenas informará os usuários sobre os riscos associados ao uso de estações de carregamento públicas, mas também provocará mudanças sistêmicas. A simulação deve despertar uma preocupação significativa entre os usuários e a opinião pública em geral, o que pode pressionar as empresas a implementar medidas adicionais de segurança em suas estações de carregamento.

As estações de carregamento são seguras? Apesar de, na maioria das vezes desconhecemos sua origem?

A privacidade dos usuários é constantemente violada, mesmo que o indivíduo acredite não ter motivos para ser alvo de um ataque. Ainda assim, sempre haverá alguém disposto a causar prejuízos.

O objetivo geral do presente artigo é recriar um ataque de *Juice Jacking*, de maneira prática, para assim compreendermos o funcionamento e informar indivíduos e organizações sobre a ameaça supracitada.

As seguintes etapas demonstram o objetivo específico deste documento para que se alcance o objetivo geral descritos no parágrafo anterior:

1. Levantamento de informações.
2. Testes de possíveis resultados.
3. Simulação referente a estrutura da estação de carregamento.

4. Configurando códigos para extrair dados de sistemas no *Raspberry pi* [1].
5. Extração de dados sensíveis para demonstração de um ataque real, com a finalidade de compreensão do leitor.

Neste artigo científico, a estrutura é organizada em seções para facilitar a compreensão e o desenvolvimento do tema. A **Introdução** apresenta o contexto do *Juice Jacking* em dispositivos Android, esclarecendo o problema de segurança e os objetivos da pesquisa. A seção de **Referencial Teórico** explora as bases conceituais dos ataques de injeção de dados e as vulnerabilidades em estações de carregamento públicas, oferecendo uma visão geral dos principais conceitos e mecanismos envolvidos. Em seguida, a seção de **Metodologia** descreve as estratégias de busca e seleção dos estudos revisados, identifica os grupos mais vulneráveis ao *Juice Jacking* e explora os primeiros passos técnicos e configurações iniciais necessárias para os experimentos. Em **Resultados**, são apresentados os detalhes do funcionamento do servidor *web* desenvolvido, incluindo as respostas obtidas quando um dispositivo móvel é conectado à porta USB do *Raspberry Pi*, a **Conclusão** resume os principais achados, destacando as contribuições do trabalho e sugerindo direções para pesquisas futuras sobre a prevenção de ataques *Juice Jacking*.

2. Referencial Teórico

Para obter os artigos e referências, foi necessário criar uma *string* de busca em diversas bases de dados, conforme fluxograma de pesquisa ilustrado na Figura 1.

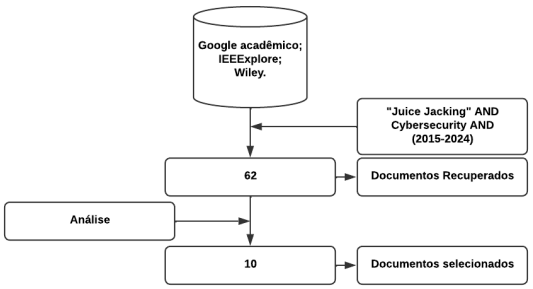

Figura 1. Fluxograma Pesquisa. Fonte: Elaborado pelo Autor

[1] O *Raspberry Pi* é um pequeno computador do tamanho de um cartão de crédito, projetado para ser acessível e fácil de usar. Ele é frequentemente utilizado em projetos de aprendizado de programação, automação e eletrônica, permitindo que usuários construam e experimentem com uma variedade de aplicações. Sua versatilidade e baixo custo o tornam popular entre estudantes e profissionais.

A *string* de busca usada foi a seguinte: *Juice Jacking AND Cybersecurity AND* (2015-2024). Dependendo da base de dados utilizada, foram realizadas algumas adaptações para ajustar a busca. A partir dessas pesquisas, vários artigos foram recuperados, e iniciamos o processo de filtragem para selecionar aqueles mais relevantes.

Conforme Singh et al. (2022) a principal forma de realizar um ataque de *Juice Jacking* é por meio de uma porta USB, pois ela permite tanto o carregamento do dispositivo quanto a transferência de dados pelo mesmo canal, conforme a Tabela 1, possibilitando o acesso não autorizado às informações dos dispositivos móveis, e tudo isso ocorre de maneira simples.

Nome do Pino	**Cor do Cabo**	**Descrição**
Pin 1	*Red*	+5V
Pin 2 D-	*White*	DATA -
Pin 3 D+	*Green*	DATA +
Pin 4	*Black*	GND

Tabela 1. Conexões USB. Fonte: Elaborado pelo autor

De acordo com Odey, Ola e Agbonlahor (2021) quando você se conecta a uma estação de carregamento e um simples *pop-up* [2] aparece em sua tela solicitando permissão de acesso a arquivos, é isso que o cibercriminoso precisa: apenas uma autorização, um simples clique, que permite ao invasor ter acesso completo a todos os dados do seu dispositivo. Ele pode realizar duas operações:

- Exfiltração: Seria a execução de códigos visando capturar dados do dispositivo e transmiti-los para fora dele, que se enquadra como furto de dados, podendo envolver fotos, vídeos, documentos e outros tipos de informações.
- Infiltração: A entrada de dados não autorizados de um agente externo, como a injeção de códigos maliciosos no dispositivo ou até mesmo a instalação de um *spyware*[3] entre as bibliotecas do aparelho, o que configura uma violação de segurança.

No entanto, como esse ataque depende da autorização da vítima, aquelas mais

[2] *Pop-up*: é uma janela gráfica que aparece repentinamente na tela de um computador ou dispositivo, geralmente em resposta a uma ação do usuário, como clicar em um *link* ou abrir um site.

[3] *Spyware* é um tipo de *software* malicioso projetado para espionar as atividades do usuário em um dispositivo, como computador ou smartphone, sem o conhecimento ou consentimento dele.

avisadas e desconfiadas podem simplesmente negar. Porém, isso não inibe o impacto dessa ameaça, como citado abaixo:

Segundo Odey, Ola e Agbonlahor (2021), o *Juice Jacking* pode ter um impacto significativo sobre indivíduos, indústrias e a sociedade, semelhante a outras ameaças de segurança, esse impacto inclui o acesso ilegal a dados e infraestruturas de TI, a implantação de bugs prejudiciais em sistemas comprometidos, e pode envolver a instalação de *malware*, negação de serviço (DoS) [4], ataques *man-in-the-middle* [5], *Phishing* [6], espelhamento de dispositivos, ataques de senha [7] e ataques *Backdoor* [8]. Além disso, esse tipo de ataque pode comprometer dispositivos móveis de maneira imperceptível, muitas vezes explorando a confiança dos usuários em estações de carregamento públicas e dispositivos compartilhados. As consequências podem ser devastadoras, desde a perda de dados pessoais sensíveis até a interrupção de operações empresariais críticas, destacando a necessidade de medidas preventivas rigorosas e adotando políticas sobre segurança cibernética.

Porém, como podemos conferir no artigo Kumar (2020) os ataques de *Juice Jacking* não ficam presos apenas em estações de carregamento, também pode ser encontrado em mouses e diversos outros dispositivos que tem cabos USB, que além de roubar dados, também podem instalar *spyware* no dispositivo conectado, se bem trabalhado, esse equipamento pode infectar até mesmo outros sistemas operacionais.

Existem diversas formas de comprometer um sistema Android, uma delas é através do Android *debug bridge*[9], que Amarante e Barros (2017) realizaram e obteveram ainda mais acessos ao sistema, até mesmo ao *Shell Unix* do aparelho conectado, e tudo isso com uma simples opção ativa no dispositivo 'Depuração USB' porém, nos sistemas mais atualizados, é necessário, também, a permissão do usuário.

[4] Negação de serviço é um ataque cibernético que visa tornar um serviço ou recurso de rede indisponível, sobrecarregando-o com tráfego excessivo ou solicitações maliciosas.

[5] *Man-in-the-middle* é um ataque cibernético onde um invasor intercepta e possivelmente altera a comunicação entre duas partes sem que elas saibam, comprometendo a segurança dos dados trocados.

[6] *Phishing* é uma técnica de fraude online onde criminosos se passam por entidades confiáveis para enganar pessoas e obter informações sensíveis, como senhas e dados bancários.

[7] Ataques de senhas envolvem tentativas de obter senhas de usuários através de métodos como força bruta, *phishing* ou engenharia social para acessar sistemas e dados protegidos.

[8] *Backdoor* é um método de contornar a segurança normal para obter acesso não autorizado a um sistema ou dados, geralmente instalado por hackers para manter controle contínuo.

[9] *Debug bridge* é uma ferramenta usada para conectar e interagir com um dispositivo, permitindo depuração, controle e execução de comandos diretamente no sistema do dispositivo.

Neste artigo, será implementada uma opção mais simples, que não requer nenhuma configuração prévia no dispositivo para acessar os dados. O processo será simplificado, onde a estação de carregamento solicitará acesso aos arquivos do dispositivo conectado, dependendo apenas da autorização do usuário.

Mediante a essas informações, esta pesquisa tem como objetivo demonstrar, de maneira prática e controlada, as técnicas utilizadas em ataques de *Juice Jacking*, revelando como esses crimes ocorrem nos bastidores. A ocorrência de um roubo ou vazamento de dados pode tornar-se difícil de ser investigada, uma vez que a origem do incidente pode permanecer desconhecida, como presente na pesquisa de Meng et al. (2015), e também podemos ver no vídeo em que a FBI avisa sobre este tipo de ataque 12News (2023) basta apenas um clique para ter suas informações roubadas. Este projeto tem como objetivo fornecer uma análise técnica da vulnerabilidade, evidenciando os métodos que cibercriminosos podem utilizar para explorar dispositivos desprotegidos. Ressaltando que essa simulação é realizada exclusivamente em um ambiente acadêmico, com o propósito de fornecer uma visão prática das ameaças, sem incentivo ao uso indevido do conhecimento gerado.

Segundo Biswal e Pani (2021) *Juice Jacking* é uma técnica de ataque cibernético na qual criminosos utilizam portas USB públicas para roubar dados ou instalar *malwares* em dispositivos conectados. Para evitar esse tipo de ameaça, algumas práticas preventivas podem ser adotadas. Uma das principais medidas é o uso de carregadores portáteis próprios (*power banks*), o que elimina a necessidade de utilizar estações de carregamento públicas. Outra prática recomendada é evitar o uso de cabos de carregamento de origem desconhecida, especialmente aqueles que já estão conectados a pontos de carregamento público, visto que esses cabos podem estar comprometidos.

Além disso, o uso de adaptadores de tomada ao invés da conexão direta via USB pode reduzir significativamente o risco de ataques, uma vez que limita a troca de dados. Outra medida eficaz envolve o uso de dispositivos conhecidos como ”*data blockers* [10], como apresentado na Figura 2, ou de cabos de carregamento que não possuem função de transferência de dados, esses dispositivos garantem que apenas a energia seja transmitida durante o carregamento, bloqueando qualquer tentativa de comunicação de dados.

[10]Um bloqueador de dados *USB* permite carregar dispositivos através de portas *USB* públicas, bloqueando a transferência de dados e protegendo contra ataques como o *Juice Jacking*.

Figura 2. ***"Data Blockers"*****. Fonte: Amazon (2013)**

Dito isso, é importante desativar a transferência automática de dados nos dispositivos, configurando-os para que utilizem apenas a função de carregamento ao serem conectados a portas USB desconhecidas. A atualização constante sobre novas técnicas de ataque e ameaças emergentes também é fundamental para manter a segurança em um ambiente tecnológico em constante evolução.

Considerando a necessidade de um ambiente prático para a simulação do ataque *Juice Jacking*, optou-se pelo uso do *Raspberry Pi 3 Model B*, um computador de placa única (SBC) amplamente utilizado em projetos de tecnologia devido ao seu baixo custo, facilidade de uso e portabilidade. A placa é equipada com um processador Broadcom BCM2837 de arquitetura quad-core ARM Cortex-A53 (ARMv8) com frequência de 1,2 GHz, oferecendo um desempenho cerca de 50% superior ao modelo anterior, *Raspberry Pi* 2. Possui 1 GB de memória RAM e suporte a conectividade de rede via *Wi-Fi* de banda única (2,4 GHz, 35 Mb/s) e *Bluetooth* 4.1, com tecnologia *BLE (Bluetooth Low Energy)*, o que proporciona maior flexibilidade em aplicações de rede e IoT. Em termos de interface, conta com uma porta Ethernet (100 Mb/s), quatro portas USB 2.0, uma porta HDMI, um conector AV de 3,5 mm, um *slot* para cartão *microSD* (utilizado para armazenar o sistema operacional), além de portas de câmera (CSI) e de exibição (DSI).

Adicionalmente, o *Raspberry Pi* 3 inclui um cabeçalho GPIO de 40 pinos, que permite a interação com diversos sensores e componentes externos, tornando-se uma ferramenta acessível e versátil para simulação de ataques de segurança e prototipagem

rápida.

Diversos estudos utilizam o *Raspberry Pi* para explorar vulnerabilidades em sistemas de dispositivos móveis e USB, como o ataque *Juice Jacking*, que explora a transferência de dados sem o consentimento do usuário ao conectar o celular em estações de carregamento comprometidas. Sua escolha é justificada pela combinação de preço acessível, flexibilidade em termos de *software* e *hardware*, e a possibilidade de criar um ambiente de teste controlado e reproduzível, características essenciais para a execução de estudos práticos como desenvolvido neste trabalho.

Com isso, o *Raspberry Pi 3B* apresenta-se como uma escolha sólida para simulação de ataques de *Juice Jacking*, permitindo a replicação de cenários reais de maneira prática, fatores que contribuíram significativamente para a sua seleção.

3. Materiais e Métodos (ou Metodologia)

Escolhemos focar no ataque de *Juice Jacking* pois, apesar de ser uma ameaça real e potencialmente devastadora, ele não recebe a devida atenção nas mídias populares. Durante uma viagem recente, observamos um grande número de pessoas utilizando estações de carregamento públicas em aeroportos e espaços comerciais, sem qualquer preocupação com possíveis riscos de segurança. Esse cenário nos fez perceber o impacto que um ataque mal-intencionado poderia ter, caso uma estação de carregamento estivesse comprometida. Além disso, o interesse pelo tema foi reforçado por uma palestra de especialistas em segurança digital, realizada na Faculdade de Tecnologia de Ourinhos, que destacou a importância de compreender e mitigar riscos desse tipo.

Os grupos mais vulneráveis a ataques de *Juice Jacking* são aqueles que permanecem desinformados ou não demonstram interesse em entender os perigos e as vulnerabilidades da tecnologia que utilizam. Essa falta de conhecimento, somada à confiança excessiva em dispositivos e locais públicos, torna essas pessoas alvos fáceis para cibercriminosos. O fator humano é um dos elementos mais frágeis da segurança digital, e, em ambientes empresariais, indivíduos desatentos podem representar um grande risco. Um comportamento negligente diante de práticas de segurança pode levar à perda de dados cruciais para uma organização, representando uma ameaça não apenas para os indivíduos, mas para toda a empresa.

O impacto no comportamento dos usuários em relação ao uso de estações de car-

regamento públicas é significativo, pois muitos desconhecem o risco envolvido. A falta de discussão sobre o *Juice Jacking* nas mídias faz com que a maioria das pessoas não tenha consciência de que carregar dispositivos em estações públicas pode representar uma ameaça à privacidade e à segurança dos dados armazenados. Se mais informações sobre esses riscos fossem amplamente divulgadas, o comportamento dos usuários provavelmente mudaria, e haveria uma redução na confiança cega depositada nessas estações de carregamento, levando-os a adotar medidas mais seguras, como o uso de bloqueadores de dados USB ou carregadores próprios.

Com base nas diferentes formas de ataques conhecidas, opta-se por utilizar um ataque específico que tem como objetivo furtar dados com a autorização da vítima, obtida através de um aceite em uma janela de *pop-up* que surgirá ao conectar o cabo USB do carregador ao dispositivo Android.

Preparação e primeiros contatos com o *Raspberry pi*: O manuseio inicial exigiu cuidados específicos para garantir tanto a integridade do *hardware* quanto um ambiente adequado para as configurações de acesso remoto. A configuração inicial contou com o uso de um tapete de borracha antideslizante e antiestático de 50x50 cm, minimizando riscos de eletricidade estática e proporcionando estabilidade durante as operações. Luvas apropriadas foram utilizadas para evitar o contato direto com os componentes eletrônicos, reduzindo a possibilidade de danos causados por descargas eletrostáticas.

Na montagem final, foram conectados ao dispositivo apenas os componentes essenciais: um carregador INOVA de 5V e 3.1A para alimentação e um cartão microSD de 64 GB, inserido na entrada inferior dedicada, para armazenar o sistema operacional. Um ventilador foi posicionado próximo à placa para manter a temperatura controlada, uma precaução importante devido à suscetibilidade da placa a aquecimentos durante o uso contínuo. Na Figura 3 foi ilustrada a estrutura frontal e traseira do *hardware*, exibindo a organização cuidadosa e o manuseio seguro durante os ajustes iniciais. A partir dessa configuração, foi possível estabelecer uma conexão via SSH, permitindo o controle remoto sem periféricos físicos, o que contribui para a segurança e a organização do espaço de trabalho.

Instalação do sistema operacional: Para o sistema operacional, optou-se pelo *Raspberry Pi OS Lite* (64 bits) – uma versão leve e sem interface gráfica, ideal para servidores *SSH* e aplicações em modo texto. Após preparar o cartão SD com um adap-

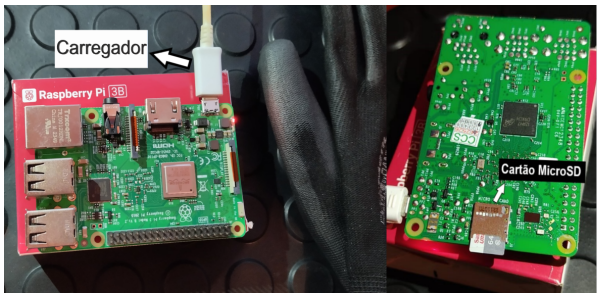

Figura 3. *Raspberry Pi* frente e verso. Fonte: Elaborado pelo autor

tador USB, foi baixado o *Raspberry Pi Imager* pelo site oficial Pi (2016) e em seguida o processo de instalação. Dentro do *Imager*, foi selecionado o modelo do nosso *Raspberry Pi* e, na categoria de sistemas operacionais, foi escolhido a versão Lite de 64 bits.

Configuração de acesso remoto e otimização: Para garantir eficiência no gerenciamento e manipulação de arquivos do dispositivo conectado, foi ativado o acesso direto via root no SSH, permitindo maior agilidade na execução de comandos administrativos. A definição da senha para o usuário root foi realizada eliminando a necessidade de prefixar comandos com **sudo** para ações elevadas.

Para implementar a função de servidor *web*, o Apache foi instalado. A instalação e configuração do PHP no Apache foram executadas e para fazer verificação de sua funcionalidade foi realizada por meio da criação de uma página de teste PHP no diretório do apache.

Gerenciamento de dispositivos USB e montagem de sistema de arquivos Android: Para detectar e acessar dispositivos conectados, o comando **lsusb** foi utilizado, revelando informações sobre todos os dispositivos USB atualmente conectados ao *Raspberry Pi*, como o cartão SD, a placa de rede *ethernet* e o próprio carregador USB.

Para gerenciar dispositivos USB e permitir que o *Raspberry Pi* detecte e monte automaticamente unidades externas (como um celular Android), instalamos o pacote **jmtpfs**, essencial para estabelecer uma conexão fluida entre o sistema e dispositivos USB.

Desafios de permissões e otimizações de acesso: Ao conectar dispositivos externos, encontramos algumas restrições de acesso, especialmente para o usuário **www-data**, que é o padrão para o Apache. Para resolver o problema, configuramos permissões avançadas, concedendo ao **www-data** privilégios administrativos de root via **visudo**, com

o objetivo de permitir o acesso irrestrito do Apache ao sistema de arquivos do dispositivo USB. Este procedimento foi feito em um ambiente seguro, onde o acesso à internet estava restrito, e foi configurado apenas para fins educacionais, visando simular ataques e testar vulnerabilidades em um ambiente controlado.

Inicialização e monitoramento de dispositivos *USB*: A princípio, a proposta era automatizar o ataque, configurando o *Raspberry Pi* para reconhecer o celular conectado e montá-lo automaticamente como uma unidade de armazenamento, permitindo o acesso imediato aos arquivos do dispositivo. Essa automação visava detectar o celular e estruturar o sistema de diretórios automaticamente, reduzindo a necessidade de intervenção manual. No entanto, dificuldades técnicas impediram a concretização dessa abordagem: ao implementar essa função, o sistema retornava erros que causavam o retorno do celular ao modo de carregamento, interrompendo o processo de acesso aos dados.

Diante desse desafio, foi adotada uma abordagem alternativa. Assim, o servidor foi configurado para monitorar dispositivos Android conectados via USB, como descrito anteriormente, exibindo a interface do servidor ao reconhecer um celular. Quando o dispositivo é conectado, surgem na tela opções de conexão, incluindo:

- Nenhuma transferência de dados (definida como padrão).
- Transferência de arquivos/Android Auto.
- Transferência de fotos (PTP)[11].

Essas opções representam diferentes níveis de acesso: ’Nenhuma transferência de dados’ permite apenas o carregamento do celular; ’Transferência de arquivos/Android Auto’ dá acesso a documentos, músicas e vídeos; e ’Transferir fotos (PTP)’ permite a importação de imagens. Apesar de parecer um processo simples, muitos usuários, no ritmo acelerado do dia a dia, acabam selecionando uma dessas opções apenas para garantir que o dispositivo esteja carregando, sem perceber o risco de expor seus dados.

Com a criação do servidor Apache, foram implementados *scripts* que permitem montar o dispositivo com um único clique. Quando o atacante, que deve estar na mesma rede que o *Raspberry Pi*, clica em ”Iniciar Montagem”, um *script* é executado para montar o celular em um formato que o sistema *Linux* reconhece e possa acessar. Isso possibilita

[11] *Picture Transfer Protocol*, é um protocolo de comunicação utilizado para transferir imagens e vídeos entre dispositivos, como câmeras digitais, *smartphones* e computadores.

a transferência de dados, permitindo que o sistema operacional comece a ler os arquivos presentes no dispositivo. O fluxograma da Figura 4 apresenta o funcionamento do servidor *Raspberry Pi* em um processo estruturado de identificação, montagem, leitura, manipulação e desmontagem dos dados de um dispositivo móvel conectado. O procedimento inicia-se com a identificação automática do dispositivo pela porta USB via comando **lsusb**. Após o reconhecimento, o sistema usa o pacote **jmtpfs** para montar o armazenamento interno do dispositivo, permitindo que o conteúdo seja acessado e manipulado como se fosse parte do sistema local.

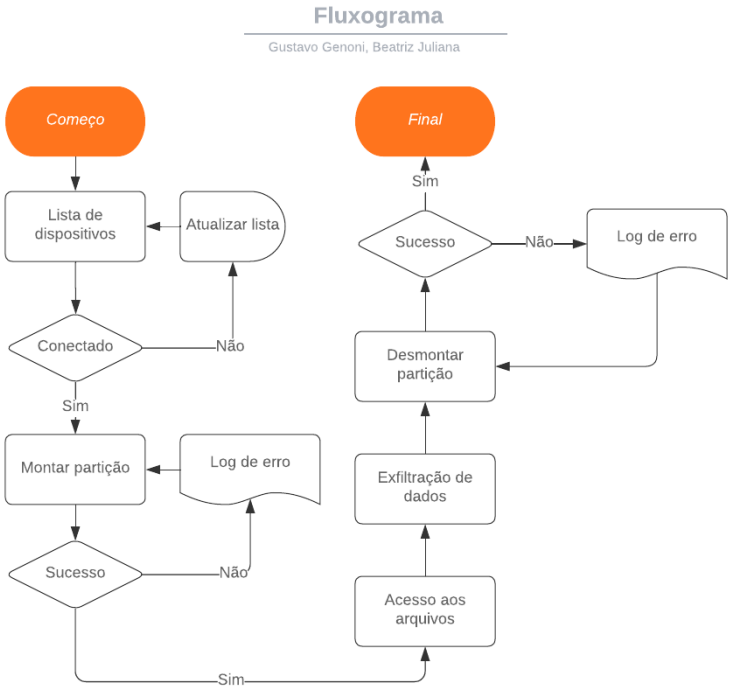

Figura 4. Fluxograma do código.Fonte: Elaborado pelo autor

Após a montagem, o servidor permite a leitura e manipulação dos dados, exibindo o conteúdo em uma interface *web* acessível. A partir dessa interface, o usuário pode visualizar, copiar ou deletar arquivos no dispositivo, sem a necessidade de periféricos adicionais. Por fim, o processo de desmontagem é acionado automaticamente para garantir a desconexão segura do dispositivo e retornar ao estado de carregamento padrão.

Esse fluxo de operações, gerenciado pelo *Raspberry Pi*, visa automatizar e simplificar o controle seguro de dados.

4. Resultados e Discussões

Foi desenvolvido um serviço *web* capaz de interagir diretamente com o sistema operacional para executar comandos específicos, utilizando uma combinação de HTML, CSS, PHP e *scripts* em Bash. O propósito prático de utilizar este servidor é proporcionar uma interface interativa para o usuário (o atacante), como podemos observar na Figura 5, facilitando o roubo de dados de celulares conectados. Através desse serviço, um usuário pode acessar uma interface *web* pelo celular e realizar ações remotamente no sistema, incluindo a execução de um ataque direcionado a um dispositivo Android conectado via USB, utilizando o protocolo MTP (*Media Transfer Protocol*).

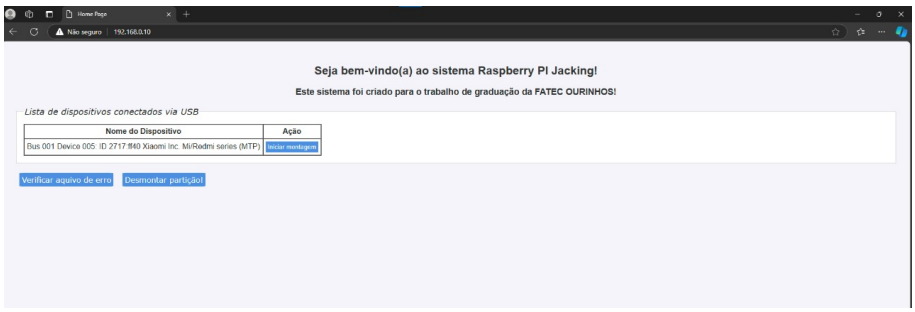

Figura 5. Interface *web*: Elaborado pelo autor

Ao estabelecer a conexão com o dispositivo Android, o sistema passa a acessar todos os diretórios e arquivos disponíveis, possibilitando tanto a exfiltração quanto a remoção de dados. O serviço permite operações com arquivos de formatos amplamente utilizados, como .pdf, .txt, .doc, .docx, .xls, .xlsx, .rtf, .odt, .ods, .ppt, .pptx, .odp, além de formatos de imagem, incluindo .jpg, .png, .jpeg, .gif, .bmp e .svg (todos os tipos de arquivos citados foram incluídos).

Ao acessar a interface *web*, que identifica automaticamente a presença de um dispositivo conectado, o usuário pode iniciar processos de cópia ou remoção de arquivos com um único clique, como podemos ver na Figura 6. No entanto, a eficiência na cópia de arquivos é um pouco lenta, pois abrange todos os arquivos contidos no diretório e subdiretórios do dispositivo, resultando em latência ao ler e obter todos os arquivos encontrados. A execução de comandos também apresentou demora, e o tratamento de erros

foi complicado, uma vez que era necessário digitar todos os comandos manualmente, o que envolvia localizar e corrigir a origem dos erros, tornando esse processo demorado. Portanto, a criação do servidor também visa facilitar o uso a partir de dispositivos móveis, otimizando a experiência do usuário.

Os arquivos copiados são armazenados localmente no *Raspberry Pi*, onde podem ser acessados diretamente pela página *web*. Dessa forma, o usuário tem a opção de visualizar e fazer *download* dos arquivos para o celular. Todo o processo ocorre de forma silenciosa para o dispositivo Android, sem exibir notificações ou alertas ao usuário do dispositivo. A única permissão necessária é o acesso aos arquivos no dispositivo Android, o que torna a operação eficaz e discreta.

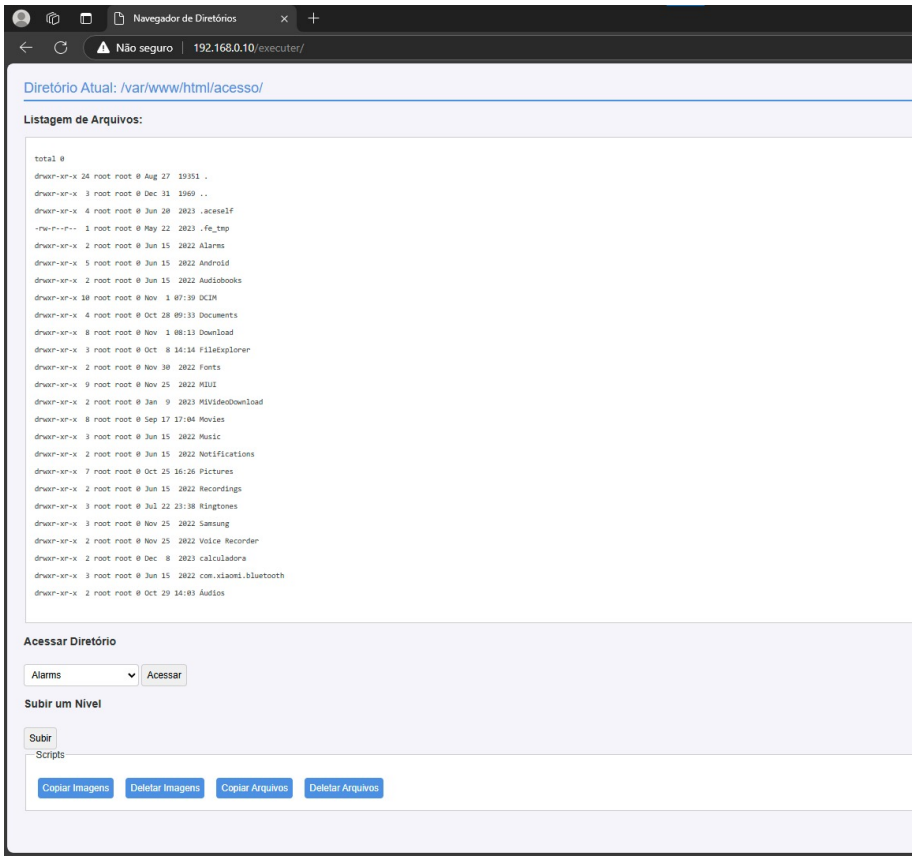

Figura 6. Arquivos encontrados: Elaborado pelo autor

Porém, é importante ressaltar que ataques desse tipo podem ser evitados por meio de medidas simples e práticas, como utilizar cabos USB que bloqueiem a transferência de dados, carregadores com suas devidas fontes, carregadores portáteis (*power banks*), ou o uso de dispositivos *data blockers*, que impedem a troca de dados enquanto o celular está carregando. Além disso, desativar o protocolo MTP em configurações do celular quando possível ou simplesmente evitar conectar dispositivos em pontos de carregamento desconhecidos ou inseguros também são práticas fundamentais. Essas medidas podem mitigar significativamente os riscos de exfiltração de dados e preservar a segurança dos usuários.

Referências

12NEWS. Fbi issues warning about juice jacking. In: . [S.l.: s.n.], 2023.

AMARANTE, J.; BARROS, J. P. Exploring usb connection vulnerabilities on android devices breaches using the android debug bridge. In: SCITEPRESS. *14th International Joint Conference on e-Business and Telecommunications, ICETE 2017*. [S.l.], 2017. p. 572–577.

AMAZON. *USB Data Blocker to Protect Against Juice Jacking - Guaranteed Charging, No Data Transfer*. 2013. Acesso em: 12 set. 2024. Disponível em: ⟨https://www.amazon.com/Blocker-Protect-Against-Guaranteed-Charging/dp/B0CBKQ5PHV⟩.

BISWAL, C. S.; PANI, S. K. Cyber-crime prevention methodology. *Intelligent Data Analytics for Terror Threat Prediction: Architectures, Methodologies, Techniques and Applications*, Wiley Online Library, p. 291–312, 2021.

KUMAR, Y. Juice jacking-the usb charger scam. *Available at SSRN 3580209*, 2020.

MENG, W. et al. Charging me and i know your secrets! towards juice filming attacks on smartphones. In: *Proceedings of the 1st ACM workshop on cyber-physical system security*. [S.l.: s.n.], 2015. p. 89–98.

NGUYEN, M. Security risks of smartphone charging in public spaces: Wired and wireless charging. 2017.

ODEY, J. A.; OLA, B.; AGBONLAHOR, I. The cyber crime of juice jacking in developing economies: Susceptibilities, consequences and control measures. *European Journal of Information Technologies and Computer Science*, v. 1, n. 5, p. 1–5, 2021.

PI, R. *Raspberry Pi 3 Model B*. 2016. ⟨https://www.raspberrypi.com/products/raspberry-pi-3-model-b⟩. Accessed: 2024-09-30.

SCHNEIER, B. *Data and goliath: The hidden battles to collect your data and control your world*. [S.l.]: WW Norton &Company, 2015.

SINGH, D. et al. Juice jacking: Security issues and improvements in usb technology. *Sustainability*, MDPI, v. 14, n. 2, p. 939, 2022.