INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CÂMPUS VOTUPORANGA

Kawam Oliveira Freitas
Pedro Henrique Veloso Gianeze

INVASÃO MOBILE

VOTUPORANGA
2020

Kawam Oliveira Freitas
Pedro Henrique Veloso Gianeze

INVASÃO MOBILE

Trabalho de Conclusão de Curso apresentado como exigência parcial para obtenção do diploma do Curso Técnico em Informática Integrado ao Ensino Médio do Instituto Federal de Educação, Ciência e Tecnologia, Câmpus Votuporanga.

Orientador: Prof. Me. Ubiratan Zakaib do Nascimento.

VOTUPORANGA
2020

FICHA CATALOGRÁFICA

Kawam Oliveira Freitas
Pedro Henrique Veloso Gianeze

INVASÃO MOBILE

Trabalho de Conclusão de Curso apresentado como exigência parcial para obtenção do diploma do Curso Técnico em Informática Integrado ao Ensino Médio do Instituto Federal de Educação, Ciência e Tecnologia, Câmpus Votuporanga.

Orientador: Prof. Me. Ubiratan Zakaib do Nascimento.

Aprovado pela banca examinadora em xx de mês de 2020.

BANCA EXAMINADORA:

_____

Prof. D.r Cicrano da Silva (para feminino use Dra.)

_____

Prof. M.e Beltrano dos Santos (para feminino use M.ª)

_____

Prof. Esp. José Luis Brasil

EPÍGRAFE

“Feliz é aquele que transfere
o que sabe e aprende o que
ensina.”
Cora Coralina

RESUMO

É inegável que os dispositivos com sistema operacional Android vêm ganhando cada vez mais adeptos com o passar do tempo, porém junto com isso o crescimento de técnicas maléficas e softwares maliciosos destinados a este sistema operacional vem se tornando cada vez maior. Os procedimentos que estes invasores utilizam conseguem ser colocados em prática pelas falhas e brechas que podem ser encontradas no sistema operacional e no próprio usuário do dispositivo. Através desta invasão, estes atacantes conseguem obter acesso há informações e dados particulares da vítima. O trabalho desenvolvido tem como objetivo expor e reportar a os usuários as principais falhas que podem ser encontradas no sistema e até mesmo as causadas pelo usuário, explicando o funcionamento do SO e como os programas funcionam dentro do sistema, e também será apresentado como um programa malicioso age quando se encontra dentro do dispositivo da vítima. Este programa malicioso vai ser um payload, que será desenvolvido utilizando a ferramenta metasploit, e posteriormente será testado em um ambiente seguro, apenas com o intuito de explicação e demonstração. Após isto, será disponibilizada uma cartilha de conscientização para os usuários sobre como manter o utilizador e seu dispositivo protegido contra ataques e ameaças.

Palavras-chaves: Segurança Mobile, Vulnerabilidade Android, Engenharia Social, Metasploit.

ABSTRACT

LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplo de *Phishing* no Twitter.....17
Figura 2 - Arquitetura do sistema operacional Android.....21

LISTA DE ABREVIATURAS E SIGLAS

OWASP - Open Web Application Security Project

SO - Sistema Operacional

APPS - Applications

APK - Android Application Pack

SI - Segurança da Informação

API - Application Programming Interface

ART - Android Runtime

HAL - Hardware Abstraction Layer

SUMÁRIO

1 INTRODUÇÃO 12
1.1 OBJETIVO GERAL 12
1.2 OBJETIVOS ESPECÍFICOS 12
1.3 JUSTIFICATIVA 13
1.4 METODOLOGIA 13
1.5 ESTADO DA ARTE 13
2 CONCEITOS BÁSICOS 15
2.1 SEGURANÇA DA INFORMAÇÃO 15
2.2 VULNERABILIDADES 15
2.3 ENGENHARIA SOCIAL 16
2.3.1 PHISHING 16
2.3.2 PRETEXTING 18
2.4 MALWARES 18
2.5 METASPLOIT 19
2.5.1 EXPLOIT 19
2.5.1.1 EXPLOIT LOCAL 20
2.5.2 PAYLOAD 20
2.6 FUNCIONAMENTO DO ANDROID E SEUS APPS 20
2.6.1 ARQUITETURA ANDROID 20
2.6.2 APK 22
3 MATERIAIS E MÉTODOS 23
3.1 APARELHOS FUNDAMENTAIS 23
3.2 SOFTWARES FUNDAMENTAIS 23
REFERÊNCIAS 25

1 INTRODUÇÃO

O trabalho tem o objetivo de mostrar o funcionamento do sistema operacional Android e seus aplicativos. Atrelado há isso, serão apresentados alguns métodos que uma pessoa mal intencionada poderá usar para conseguir ter acesso ao celular da vítima sem seu consentimento. Para que isso seja possível um código malicioso (payload) vai ser desenvolvido e testado em um ambiente controlado sem fins prejudiciais a terceiros, apenas como prova de conceito. Após todos os testes serem feitos e apresentados, será elaborada e disponibillizada uma cartilha de prevenção com o intuito de conscientizar os usuários de dispositivos Android para que eles não caiam e se atentem sobre este tipo de golpe.

1.1 OBJETIVO GERAL

Expor o funcionamento do sistema operacional Android e seus aplicativos, após isso será desenvolvido um payload em ambiente controlado como prova de conceito, para mostrar seu funcionamento. Ao final será apresentado e disponibilizado uma cartilha de prevenção, para que os usuários não caiam neste tipo de ataque e se mantenham seguros.

1.2 OBJETIVOS ESPECÍFICOS

- Aprofundar conhecimentos sobre Segurança da Informação.
- Estudar sobre os sistemas operacionais linux e Android.
- Estudar as ferramentas que serão utilizadas.
- Disponibilizar aos usuários uma cartilha de segurança que os auxiliem a manter seu dispositivo Android e a si mesmo protegido.
- Informar a os usuários os principais tipos de ataques de engenharia social com os quais eles podem se deparar.

1.3 JUSTIFICATIVA

Com o crescente número de usuários do sistema operacional Android, a quantidade de softwares maliciosos e técnicas de ataque para este SO sofreu um aumento significativo nos últimos anos (KASPERSKY, 2014). Infelizmente nem todos os usuários estão cientes das informações apresentadas e por isso não tomam os devidos cuidados com seus dispositivos, se tornando assim alvos mais fáceis para os criminosos. Tendo em vista as informações apresentadas o objetivo principal deste trabalho é conscientizar os usuários de dispositivos Android os riscos que eles e seus aparelhos estão expostos.

1.4 METODOLOGIA

Para atingirmos o objetivo principal deste projeto, utilizaremos as seguintes metodologias:

- Levantamento Bibliográfico: Este procedimento será utilizado para nos auxiliar na utilização das ferramentas necessárias para a conclusão do projeto.
- Pesquisa Exploratória: Por meio dessa análise será possível ter acesso a todos os dados que foram obtidos através das técnicas de engenharia social. Graças a este processo vamos conseguir reunir todos os problemas encontrados durante a pesquisa;
- Pesquisa Explicativa: Última fase, onde vamos registrar todos os dados e por meio dos problemas e das soluções obtidas chegaremos em uma solução incontestável para o problema apresentado.

1.5 ESTADO DA ARTE

Diversas abordagens já foram feitas sobre o tema invasão Android, porém utilizando ferramentas e metodologias diferentes.

No trabalho de Mendes (2017), ele utiliza uma abordagem diferente com a entidade OWASP que é um projeto aberto, o qual é mantido pela comunidade. O seu objetivo é criar documentos, ferramentas, tecnologias e metodologias para manter a segurança de aplicações. Um dos projetos oferecidos pela entidade é o OWASP

Mobile Security Protect, que é responsável por oferecer recursos a desenvolvedores de aplicativos mobile para mantê-los seguros. O objetivo deste trabalho é detectar e analisar vulnerabilidades encontradas em aplicações mobile seguindo critérios da OWASP com o uso de ferramentas de *pentesting*.

Já no projeto de Junior (2016), foi utilizada a ferramenta MSFVENOM, com o intuito de realizar o ataque em um dispositivo com o sistema operacional Android na versão 3.0. Neste trabalho é visada a utilização do *framework* Metasploit para o desenvolvimento da ferramenta MSFVENOM, logo após um antivírus foi instalado no sistema operacional para detectar o impacto do software diante de um ataque deste nível.

No estudo realizado por Almeida (2013), foi constatado que com o crescente número de usuários do sistema operacional Android, a quantidade de vulnerabilidades encontradas vem crescendo de forma incansável. Levando isso em consideração, foi realizada um levantamento e uma análise sobre as vulnerabilidades encontradas na plataforma Android, tendo em vista todas essas falhas, foram pesquisadas e testadas ferramentas de proteção para ajudar a assegurar a segurança do sistema operacional.

O trabalho elaborado por Della Flora (2010) tem o intuito de evidenciar o uso do *framework* Metasploit no desenvolvimento de um *exploit* para realizar a exploração de vulnerabilidades. A vulnerabilidade explorada neste trabalho é denominada *buffer overflow*, que consiste em uma falha de segurança na qual se utiliza o estouro de dados, sobrecarregando uma variável do sistema.

Assim como foi constatado por Almeida (2013), Moura (2017) dá seguimento a essa linha de raciocínio, confirmando que com o crescente número de usuários de celulares com o sistema operacional Android, cada vez mais vai ampliar-se as falhas e vulnerabilidades encontradas no sistema. O objetivo do trabalho feito por Moura foi realizar uma vasta pesquisa para encontrar e detalhar o melhor *payload* para realizar ataques ao sistema operacional Android, e com isso são gerados *backdoors* com a ferramenta MSFVENOM e uma comparação foi feita entre eles.

2 CONCEITOS BÁSICOS

2.1 SEGURANÇA DA INFORMAÇÃO

Alves (2006, p.15), aponta a segurança da informação como uma técnica de proteção de dados e informações confidenciais que não podem ser acessados por qualquer pessoa.

Segundo Dantas (2011, p.11), os pilares básicos da segurança da informação são:

- Confidencialidade: Garantir que nenhuma informação seja acessada ou divulgada sem permissão.
- Integridade: Garantir com que os dados e as informações não possam ser manipulados de forma prejudicial por pessoas não autorizadas.
- Disponibilidade: Garantir que os dados ou informações desejadas possam ser acessados a qualquer momento por pessoas autorizadas.

Toda informação possuei um ciclo de vida, um tempo de vida útil onde podem ser armazenadas, transportadas e por fim destruídas - colocando um fim na vida útil da informação (Beal, 2005).

2.2 VULNERABILIDADES

Vulnerabilidade é uma falha de segurança que permite um atacante ter acesso a informações e a dados sensíveis, elementos esses os quais ele não poderia ter acesso.

De acordo com Tripla (2017):

> Em termos do léxico, algo vulnerável é aquilo que é sensível a determinadas ameaças. É uma “brecha” em algo que permite um ataque ou associa-se a determinado risco.

Entre os tipos de vulnerabilidades existentes podemos destacar duas, a humana e a de *software*. A vulnerabilidade humana consiste no ato do usuário conceder permissões para um aplicativo mal intencionado no sistema, realizando o *download* e/ou a instalação do programa. Já a vulnerabilidade de *software* pode ser causada por um SO desatualizado e/ou mal protegido.

2.3 ENGENHARIA SOCIAL

O engenheiro social utiliza das variadas técnicas presentes na Engenharia Social para enganar as pessoas e retirar delas informações que lhe são úteis. Mitnick e Simon, referências na área, salientam que:

> Engenharia Social usa a influência e a persuasão para enganar as pessoas e convencê-las de que o engenheiro social é alguém que na verdade ele não é, ou pela manipulação. Como resultado, o engenheiro social pode aproveitar-se das pessoas para obter as informações com ou sem o uso da tecnologia (MITNICK, SIMON, 2003, p.6).

Para esses autores, o engenheiro social utiliza de meios bem simples para enganar as pessoas, um telefone ou até mesmo à *Internet*, fazendo com que elas cedam informações confidenciais ou que quebrem algum protocolo de segurança instituído pela empresa. Quando o engenheiro social faz uso das técnicas de Engenharia Social, ele está se aproveitando da tendência humana em confiar nas pessoas e para se aproveitar disso ele leva em consideração o princípio básico da Engenharia Social: o ser humano é o elo mais fraco dentre os mecanismos de segurança.

Portanto, pode-se dizer que não importa quão boa sejam seus mecanismos de segurança, se a organização não investir em treinamento humano, ou seja, capacitar a equipe para conseguir identificar e solucionar ou ao menos diminuir o impacto de possíveis ataques, a organização ainda corre um grande risco.

Há uma grande gama de trabalhos que debatem sobre o ser humano ser “o elo mais fraco” na segurança da informação e os riscos que isso pode trazer às corporações caso não forneçam um treinamento adequado e eficiente.

2.3.1 PHISHING

O termo *phishing* pode ser traduzido como pescaria e é atualmente o tipo mais comum de engenharia social utilizado pelos criminosos do mundo todo, devido ao fato de ser barato e conseguir atingir até milhões de contas (LEITE, PEREIRA, 2019). Estes criminosos utilizam *spams*, *websites* maliciosos, mensagens instantâneas e email como ferramenta para que as pessoas forneçam informações sigilosas, como

números de contas bancárias e de cartões de crédito. A Figura 1 abaixo é um exemplo de *phishing*, representado por uma oferta de cartões de crédito personalizados.

Figura 1 – Exemplo de *Phishing* no Twitter

Fonte: WeLiveSecurity (2019).

Após ver o exemplo acima, pode-se concluir que o *phishing* faz-se passar por um comunicado, notícia ou na maioria dos casos anúncios ou *websites* de fontes credíveis, focalizando o envio destes para usuários desinformados sobre o assunto. Outros tipos de *phishing* são bem conhecidos e usados, como o *smishing* e o *vishing.*

De acordo com a Kaspersky, em 2018 os brasileiros foram os que mais sofreram ataques de *phishing* no mundo. Este levantamento mostrou que no ano de 2018 cerca de 23% dos usuários do país sofreram ataques e em 2017 esta estatística sobe para 30% (KASPERSKY, 2018). Isso ocorre pelo fato de que uma parcela esmagadora das pessoas na sociedade não possui o conhecimento necessário sobre

Segurança da Informação para se proteger de ataques efetuados pelos engenheiros sociais, assim se tornando presas fáceis para esses criminosos.

Outros tipos de *phishing* são bem conhecidos e usados, como o *smishing* e o *vishing*. O *vishing*, segunda a Avast (2020), é uma técnica que consiste no formato via SMS do *phishing,* ao realizar o que se pede, a vítima vai executar um *malware* em seu aparelho, que poderá capturar suas informações pessoais e enviá-las ao criminoso. O *vishing* é a abreviação de "*phishing* por voz", é a forma de áudio do *phishing* na *web*. Na chamada telefônica os criminosos se passam por outras pessoas confiáveis, desse modo criam um vínculo com a vítima, o que torna o processo de extração da informação mais rápido e eficaz (AVAST, 2018).

2.3.2 PRETEXTING

O termo *pretexting* é originário da palavra "pretexto", o cenário inventado pelo criminoso, que geralmente se passa por um funcionário real para passar mais credibilidade e confiança ao alvo, têm o objetivo de fazer com que o alvo não se sinta desconfortável ou tenha suspeitas de que seja um golpe, dessa forma ele realizará, sem grandes problemas, o desejo do engenheiro social (BAER (2008) apud Silva (2013)).

O intuito de todas as técnicas de Engenharia Social se baseiam em retirar informações confidenciais de um indivíduo para fins ilegais, o que muda é apenas qual técnica será utilizado.

2.4 MALWARES

Segundo AVG (2019), *Malware* é um *software* malicioso podendo ser um aplicativo, *script* ou código que possui o objetivo de danificar e/ou prejudicar o sistema ou até mesmo o *hardware* em que o SO encontra-se instalado.

AVG (2019), ainda esclarece que nenhum dispositivo conectado a internet está protegido, incluindo o Android:

> PCs não são os únicos dispositivos que são infectados por malwares: qualquer dispositivo que pode ser conectado à internet corre esse risco, e isso inclui smartphones Android. Mesmo que você não escute muito falar sobre eles, ataques a Androids são cada vez mais comuns, com sites tipo phishing, aplicativos falsos e lojas de aplicativos não oficiais sendo os principais distribuidores de softwares perigosos.

Esse tipo de arquivo pode ser instalado ou executado como um programa, fazendo com que assim o sistema seja infectado. Esses *softwares* maliciosos também podem se espalhar pela rede, ou até mesmo vir atrelados a um programa supostamente confiável.

2.5 METASPLOIT

De acordo com MAYNOR (2007), HD Moore deu início a codificação da ferramenta em 2003, inicialmente desenvolvida na linguagem de programação Perl. O objetivo inicial do Metasploit era realizar testes em redes. Depois do lançamento, o projeto sofreu várias mudanças e foi se tornando gradualmente um *framework*, logo após estas mudanças a ferramenta foi totalmente reescrita na linguagem de programação Ruby. Depois de se tornar oficialmente um *framework* de código livre, o seu principal objetivo se tornou desenvolver e configurar *exploits* e módulos auxiliares com o intuito de explorar falhas e vulnerabilidades.

2.5.1 EXPLOIT

Exploits são arquivos que possuem um código malicioso dentro deles. Existem vários tipos, mas cada um possui uma maneira diferente de atuar. Entre os *exploits* existentes podemos detacar o *exploit* local, o qual foi escolhido para o desenvolvimento do *payload*.

Kaspersky (2016), ainda afirma que os exploits são um subconjunto de *malware* podendo ser eles locais ou remotos:

> Exploits são um subconjunto de malware. Normalmente, são programas maliciosos com dados ou códigos executáveis capazes de aproveitar as vulnerabilidades de sistemas em um computador local ou remoto.

2.5.1.1 EXPLOIT LOCAL

Exploram vulnerabilidades e permissões de sistemas com o intuito de conseguir acesso ao usuário root (administrador) do *software* atacado. Esta técnica consiste em obter acesso ao shell do sistema, e executar um *script*. A execução deste exploit somente se torna possível se os dispositivos estiverem conectados na mesma rede.

2.5.2 PAYLOAD

O *payload* é um código malicioso que é infiltrado por um *exploit* em uma falha ou brecha do sistema atacado. Nele contém as instruções que vão ser executadas após ele ser inserido no sistema.

De acordo com MAYNOR (2007), payload é uma instrução de códigos programadas em Assembly e permitem um atacante ter acesso remoto sobre um dispositivo.

2.6 FUNCIONAMENTO DO ANDROID E SEUS APPS

2.6.1 ARQUITETURA ANDROID

O Android é um sistema operacional da Google, que foi desenvolvido baseado em Linux. O SO é destinado apenas para dispositivos móveis. O Android é composto por 6 camadas de *software*, são elas:

O *kernel* do Android é o núcleo do sistema, em seu desenvolvimento ele foi baseado no Linux 2.6. O *kernel* é responsável por fazer o intermédio entre o hardware e o software do dispositivo móvel, fazendo com que assim seja possível realizar ações como gerenciamento de memória RAM/ROM, processador, controle de processos executados no sistema e gerenciamento de *drivers (*Brahler, 2010).

No momento em que uma *framework* API aciona o *hardware* do dispositivo, a HAL é responsável por carregar módulos que sejam compatíveis e que possibilitem a configuração do hardware (Developer Android, 2020).

O Android *Runtime* é a camada responsável por executar várias máquinas virtuais Dalvik nos dispositivos, para toda aplicação executada no Android é criada uma máquina virtual (VM) para que não haja nenhuma interferência na execução nos processos do sistema. A ART também é responsável por simplificar o gerenciamento e diminuir o uso de memória no sistema *(*Brahler, 2010).

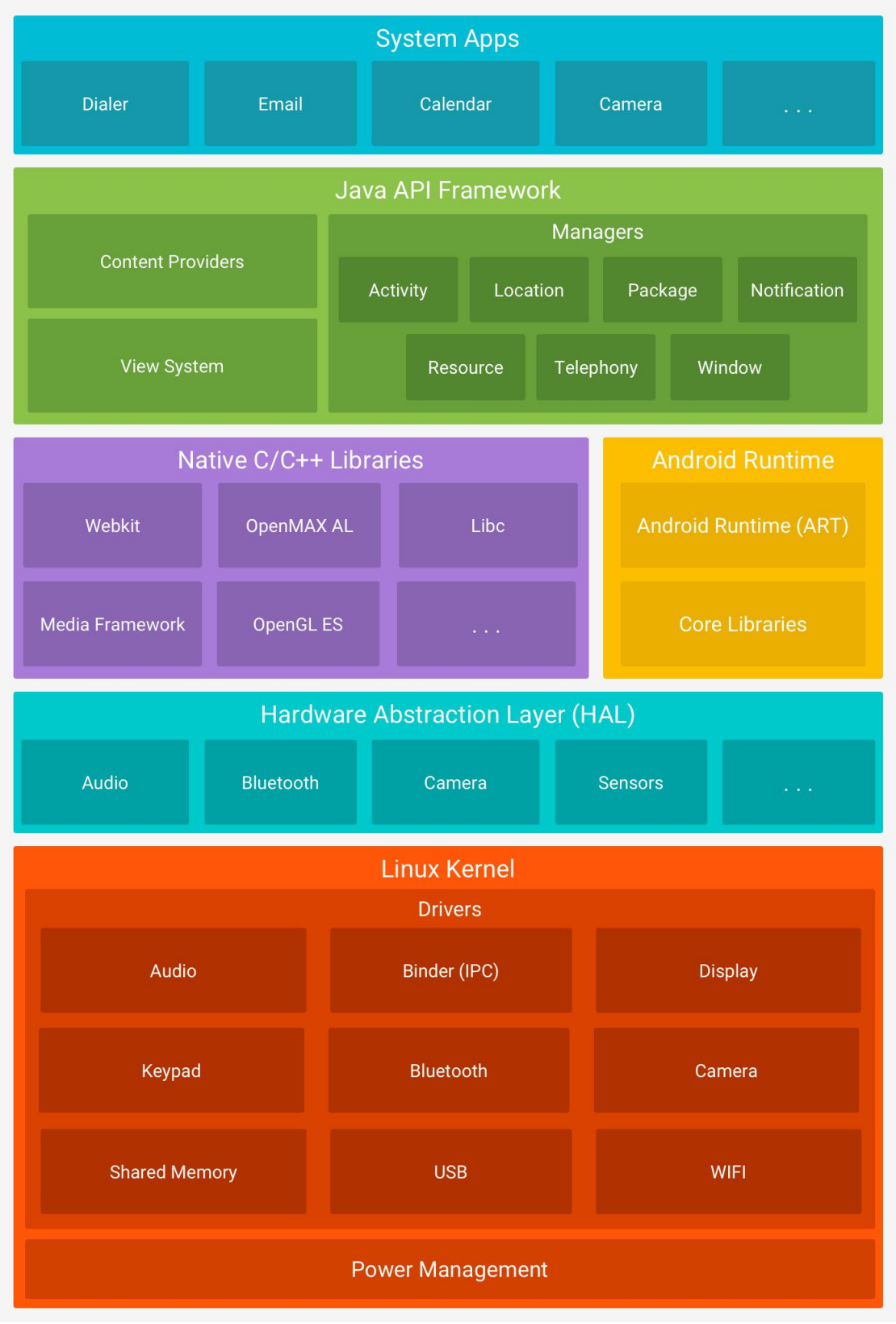
As bibliotecas C/C++ do Android são responsáveis por permitir o funcionamento correto de várias ferramentas do sistema, como a ART. Com elas também é possível visualizar imagens e objetos 2D ou 3D, fontes bitmap e vetorizadas e realizar acessos ao banco de dados SQLite (banco de dados do Android). Várias funcionalidades disponíveis em navegadores web só são executadas por conta dessas bibliotecas *(*Brahler, 2010).

Na camada de *framework* está disponível todo um conjunto de ferramentas e instrumentos que são necessários para o desenvolvimento de aplicativos para o Android, fazendo com que a utilização desses módulos fique mais simplista.

A camada de aplicação é onde fica contido a interface do usuário final e todos os aplicativos do sistema.

Como Podemos ver na figura 2 estão representadas todas as camadas do sistema operacional Android e seus componentes.

Figura 2 - Arquitetura do sistema operacional Android



Fonte: Android Developers (2020).

2.6.2 APK

Conforme Brahler (2010) explica é possível concluir que, o acrônimo “APK” vem da expressão em inglês *Android Application Pack*. Um APK é um arquivo compactado responsável por armazenar todas as informações necessárias para a instalação de um aplicativo, este tipo de arquivo encontra-se disponível apenas para dispositivos que possuem o sistema operacional Android.

Um aplicativo contendo a extensão “.apk” pode ser baixado diretamente da loja de aplicativos oficial do Android, a Google Play, porém este mesmo tipo de arquivo também pode ser encontrado em fontes não oficiais. Um aplicativo baixado de uma fonte não confiável, ou seja, fora da Google Play possui altas chances de não ser legítimo e possuir algum tipo de *malware* executando junto com ele.

Para evitar com que outros aplicativos fora da loja oficial fossem instalados em dispositivos com o sistema operacional Android, o próprio sistema e a Google Play disponibilizam uma ferramenta que impossibilita a instalação de aplicativos de outras fontes, porém todas essas funções de proteção podem ser facilmente desabilitadas pelo usuário caso necessário.

3 MATERIAIS E MÉTODOS

3.1 APARELHOS FUNDAMENTAIS

- Uma roteador com conexão a rede.
- Um aparelho celular Android.

3.2 SOFTWARES FUNDAMENTAIS

- Metasploit
- Máquina virtual com o sistema operacional Kali Linux 64bits.

REFERÊNCIAS

MENDES, Dimas Albuquerque. **Análise de Vulnerabilidades em Aplicações Android com o Uso de Ferrramentas de Teste de Intrusão e a Metodologia OWASP**. 2017.

JUNIOR, Eliezer de Souza Batista. **Uso de vírus desenvolvido no software MSFVENOM contra sistemas operacionais Android com utilização de mensagem SMS**. 2016.

ALMEIDA, Josiane. **ANÁLISE DA SEGURANÇA E DE FERRAMENTAS NA PLATAFORMA ANDROID.** 2013.

DELLA FLORA, Julio Cesar Liviero. **Desenvolvimento e aplicação de exploits utilizando o metasploit framework**. Revista Terra & Cultura: Cadernos de Ensino e Pesquisa, v. 26, n. 51, p. 113-124, 2018.

De Melo, L. P., Amaral, D. M., Sakakibara, F., de Almeida, A. R., de Sousa Jr, R. T., & Nascimento, A. (2011). **Análise de Malware: Investigação de Códigos Maliciosos Através de uma Abordagem Prática**. Minicursos do XI Simpósio Brasileiro de Segurança daInformação e de Sistemas Computacionais (SBSeg).

KASPERSKY. **O problema crescente do Malware no Android**. Disponível em: https://www.kaspersky.com.br/blog/o-problema-crescente-do-malware-no-android/3044/. Acesso em: 20 set. 2020.

ALVES, Gustavo Alberto. **Segurança da Informação: uma visão inovadora da gestão**. Rio de Janeiro: Ciência Moderna Ltda, 2006.

DANTAS, Marcus Leal. **SEGURANÇA DA INFORMAÇÃO: UMA ABORDAGEM FOCADA EM GESTÃO DE RISCOS**: Disponível em: http://www.marcusdantas.com.br/files/seguranca_informacao.pdf#page=13&zoom=100,80,94. Acesso em: 20 set. 2020.

BEAL, Adriana. **Segurança da informação: princípios e melhores práticas para a proteção dos ativos de informação nas organizações**. São Paulo: Atlas, 2005.

BRASIL. Decreto n° 4.553, de 27 de novembro de 2002. Diário Oficial da União, poder Executivo, Brasília, 30/12/2002.

TRIPLA. **Entendendo melhor uma Vulnerabilidade**. Disponível em: https://triplait.com/-que-e-vulnerabilidade-na-area-de-ti/. Acesso em: 20 set 2020.

MITNICK, K., SIMON, W., WOZNIAK, S. **"The Art of Deception"**, John Wiley & Sons. 2002.

MITNICK, Kevin D.; SIMON, William L. A arte de enganar. **Ataques de Hackers: Controlando o Fator**, 2003.

LEITE, Iago Piccoli; PEREIRA, Fagner Coin. ENGENHARIA SOCIAL. **Seminário De Tecnologia Gestão E Educação**, v. 1, n. 2, p. 11-14, 2019.

RODRIGUES, Renato. Brasil é o País com mais usuários atacados por phishing. **Kaspersky**, 2019. Disponível em: https://www.kaspersky.com.br/blog/brasil-ataques-phishing/11826/ Acesso em 28 mai. 2020.

RODRIGUES, Renato. Brasileiros são maiores vítimas de golpes phishing no mundo. **Kaspersky**, 2018. Disponível em: https://www.kaspersky.com.br/blog/phishing-klsec-brasil-assolini/10642/. Acesso em 08 jun. 2020.

AVAST. c-phishing. **Avast.** Disponivel em: https://www.avast.com/pt-br/c-phishing. Acesso em 08 jun. 2020.

AVAST. c-pharming. **Avast.** Disponivel em: https://www.avast.com/pt-br/c-pharming#topic-4. Acesso em 08 jun. 2020.

AVAST. o-spoofing. **Avast.** Disponivel em: https://www.avast.com/pt-br/c-spoofing. Acesso em 16 jun. 2020.

AVG. **O que é malware? Como malwares funcionam e como se livrar deles**.

Disponível em: https://www.avg.com/pt/signal/what-is-malware. Acesso em: 20

set. 2020.

KASPERSKY. **O problema crescente do Malware no Android**. Disponível em: https://www.kaspersky.com.br/blog/o-problema-crescente-do-malware-no-android/3044/. Acesso em: 20 set. 2020.

MAYNOR, David. **Metasploit toolkit: for penetration testing, exploit development, and vulnerability research**. Syngress.2007.

DEVELOPER ANDROID. **Arquitetura da plataforma**. Disponível em: https://developer.android.com/guide/platform?hl=pt-br. Acesso em: 20 set. 2020.

BRAHLER, **Stefan. Analysis of the Android Architecture**:06/jun./2010. Disponível em: https://os.itec.kit.edu/downloads/sa_2010_braehler-stefan_android-architecture.pdf. Acesso em: 20 set. 2020.