INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-GRANDENSE *CAMPUS* PASSO FUNDO
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET

JOSIANE ALMEIDA

ANÁLISE DA SEGURANÇA E DE FERRAMENTAS NA PLATAFORMA ANDROID

Prof. Me. Lisandro Lemos Machado

PASSO FUNDO, 2013

JOSIANE ALMEIDA

ANÁLISE DA SEGURANÇA E DE FERRAMENTAS NA PLATAFORMA ANDROID

Monografia apresentada ao Curso de Tecnologia em Sistemas para Internet do Instituto Federal Sul-Rio-Grandense, *Campus* Passo Fundo, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet, sob orientação do Prof. Me. Lisandro Lemos Machado.

PASSO FUNDO, 2013

JOSIANE ALMEIDA

ANÁLISE DA SEGURANÇA E DE FERRAMENTAS NA PLATAFORMA ANDROID

Trabalho de Conclusão de Curso aprovado em ____/____/____ como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet

Banca Examinadora:

_____

Orientador Prof. Me. Lisandro Lemos Machado

_____

Convidado Prof. Esp. José Antônio Oliveira de Figueiredo

_____

Convidado Prof. Me. Carlos Alberto Petry

_____

Prof. Me. Evandro Miguel Kuszera
Coordenação do Curso

PASSO FUNDO, 2013

Aos meus pais,
irmãos, irmã e familiares
pelo incentivo e força
em todas as horas.

AGRADECIMENTOS

Agradeço aos meus queridos pais José Ubirajara Silva Almeida e Maria de Lourdes Almeida, pelo carinho, apoio e por me ensinarem valores que levarei para toda a minha vida. Aos meus irmãos José, Jeferson, á minha irmã Andressa e á minha cunhada Juliana, que sempre me apoiaram em minhas decisões e me deram suporte não só para finalizar esta monografia, mas também por enfrentar outros desafios em minha vida.

À minha família em geral, que sempre me motivaram a ser uma pessoa melhor, ás minhas avós, Alzira e Bega, que demonstram que na vida devemos ser perseverante e sempre acreditar em nossos princípios.

Aos funcionários do Instituto Federal Sul-Rio-Grandense, em especial á Rose, que sempre esteve á disposição para ajudar com a parte burocrática e também á Silvana, sempre simpática e atenciosa com os alunos.

Aos professores do IFSUL, desde o curso técnico de Sistemas de Informação até o curso Superior de Tecnologia em Sistemas para a Internet, agradeço por todo o conhecimento que adquiri nesta longa jornada. Agradeço ao meu orientador Me. Lisandro Lemos Machado, pela sua dedicação e empenho na realização desta monografia. Ao professor Juliano Menegaz (in memoriam), que sempre trazia uma mensagem de motivação e um sorriso no rosto.

Aos meus colegas com quem passei bons momentos em toda a minha trajetória no IFSUL, Valdecir, Paulo, Algazir, Kléber, Ana Camila e Giuseppe.

“A mente que se abre a uma
nova ideia jamais voltará ao
seu tamanho original.”
Albert Einstein

RESUMO

A utilização de dispositivos móveis cresceu significativamente nos últimos anos, sendo a Plataforma Android uma das Plataformas mais utilizadas nestes dispositivos. Porém com o aumento desta utilização, também aumentam as vulnerabilidades, tornando a Plataforma mais visada para a realização de ataques por pessoas mal intencionas, mas muitos destes usuários acreditam que estão protegidos e de que seus dispositivos estão seguros de ataques, no entanto, os atacantes estão cada vez mais presentes na, criando novas formas de ataques, como a criação de aplicações maliciosas, utilização de redes públicas para ataques via wi-fi, além de muitos outros. Para que os dispositivos possam estar seguros, é necessária a utilização de ferramentas de segurança, porém estas ferramentas possuem diferenças de serviços disponibilizadas aos usuários, podendo uma possuir mais opções de segurança do que outra. No presente trabalho, foi realizada uma pesquisa sobre as vulnerabilidades presente na Plataforma Android, e também algumas ferramentas de segurança, para que estes ataques não sejam realizados. Para a realização dos testes quanto à ferramenta mais eficiente, foi utilizado o Sistema Operacional Backtrack para a simulação de alguns ataques ao dispositivo, e também a utilização das ferramentas quanto ao seu custo computacional, os pontos positivos quanto ao que as ferramentas possuem para a utilização de seus usuários, bem como suas deficiências. Após foi realizado uma comparação dentre as ferramentas, chegando ao resultado de qual obteve melhor desempenho total e a menos eficiente.

Palavras-chave: Segurança; Dispostivos Móveis; Plataforma Android; Ferramentas.

ABSTRACT

The use of mobile devices has grown significantly in recent years, and the Android platform one of the most used platforms these devices. However, with the increasing use of this also increases the vulnerability, and the platform to perform more targeted attacks by malicious people, but many users are protected and believe that their devices are secure from attack, however, attackers are increasingly present in the Android platform, creating new forms of attacks, such as the creation of malicious applications, use of public networks to attacks via wi-fi, plus many others. To which devices can be sure, it is necessary to use security tools, but these tools have differences in services available to users, one can have more security options than another. In this study, a survey was conducted on the vulnerabilities present in the Android platform, and also some security tools, so that these attacks are not carried out. For the tests as to the most effective tool was used Operating System Backtrack to simulate some attacks to the device, and also the use of the tools on the computational cost, the positives about what tools they have to use of its members, as well as its shortcomings. After a comparison was made among the tools, arriving at the result of which had the best overall performance and less efficient.

Keywords: Security; Mobile Devices; Platform Android; Tools.

LISTA DE TABELAS

Tabela 1 - Informações dos dispositivos móveis 54
Tabela 2 - Configuração da máquina virtual 54
Tabela 3 - Resultado dos testes - AVG Antivírus Free 66
Tabela 4 - Resultado dos testes - Lookout Security & Antivírus 69
Tabela 5 - Resultados dos testes - Dr.Web Antivírus Light 71
Tabela 6 - Resultados dos testes - Avast! Mobile Security 73
Tabela 7 - Resultados testes - Norton Mobile Security 75
Tabela 8 - Resultados testes - ESET Mobile Security 77
Tabela 9 - Resultados testes - DroidWall 80
Tabela 10 - Resultados testes - Root Firewall 82
Tabela 11 - Resultados dos testes - Avast Mobile Security & Antivírus- Firewall 86
Tabela 12 - Resultados dos testes Hideman VPN 89
Tabela 13 - Resultados dos testes - Viatun 4 VPN 90

LISTA DE FIGURAS

Figura 1 - Arquitetura do Android 22
Figura 2 - Sistema de criptografia 33
Figura 3 - Sistema de Criptografia Simétrica 34
Figura 4 - Sistema de Criptografia Assimétrica 34
Figura 5 - Modelo do arquivo AndroidManifest.xml 39
Figura 6 - Imagem do Aplicativo aCalendar 40
Figura 7 - Ataque á webView por páginas web 43
Figura 8 - Ataque á Web View de aplicativos mal-intencionados 43
Figura 9 - Fases do teste de invasão 48
Figura 10 - Configuração do arquivo rfcomm.conf 55
Figura 11- Comando scan 56
Figura 12 - Comando rfcomm 56
Figura 13 - Tela ferramenta Bluediving 57
Figura 14 - Dados do dispositivo atacado 57
Figura 15 - Resultado do ataque realizado por comando AT 58
Figura 16 - Ataque arpspoof 59
Figura 17 - Imagem driftnet 60
Figura 18 - Imagem da tela inicial da ferramenta metasploit 60
Figura 19 - Módulos encontrados para Plataforma Android 61
Figura 20 - Módulo usado no ataque 61
Figura 21 - Comandos da máquina do atacante 61
Figura 22 - Máquina do atacante esperando ser acessado 62
Figura 23 - Resultado do ataque realizado 62
Figura 24 - Backup realizado pela ferramenta Titanium 63
Figura 25 - Arquivos extraídos do aplicativo 63
Figura 26 - Arquivo sem alteração 64
Figura 27 - Arquivo com alteração 64
Figura 28 - Menu da Ferramenta AVG 68
Figura 29 - Menu do Botão Proteção 68
Figura 30 - Menu do Botão Antifurto 68
Figura 31 - Menu do Botão Proteção 68
Figura 32 - Menu do Botão Privacidade 68
Figura 33 - Janela para escolha de arquivo para verificação 68
Figura 34 - Menu do Botão Privacidade 68
Figura 35 - Imagem do site da AVG para bloqueio de dispositivo 68
Figura 36 - Menu Principal ferramenta Lookout 70
Figura 37 – Página do botão de 70
Figura 38 – Página do Botão Backup 70
Figura 39 – Página do Botão 70
Figura 40 – Página de Dispositivo Perdido 71

Figura 41 - Menu Inicial da ferramenta Dr. Web antivírus 72
Figura 42 – Página com todos 72
Figura 43 – Página da opção Custom scan 72
Figura 44 – Estatística da ferramenta 72
Figura 46 - Menu Inicial da ferramenta Avast! 74
Figura 47 – Menu Inicial daferramenta Avast! 74
Figura 48 – Página Escaneamento 74
Figura 49 – Anti-Theft do Avast! 74
Figura 50 - Gerenciador de Programas 74
Figura 51 – Página de Privacidade 74
Figura 52 - Menu principal da ferramenta Norton Mobile Security 76
Figura 53 – Página do botão 76
Figura 54 – Página Anti-Theft 76
Figura 55 –Página com serviços 76
Figura 56 - Menu principal da ferramenta Eset 78
Figura 57 – Menu Inicial da 78
Figura 58 – Página AntiSpam Eset 78
Figura 59 – Antifurto Eset 78
Figura 60 - Gerenciador eset 78
Figura 61 - Modo de escolha da lista Droidwall 81
Figura 62 - Seleção dos aplicativos bloqueados Droidwall 81
Figura 63 - Menu do Droidwall 81
Figura 64 - Script Customizado Droidwall 81
Figura 65 - Bloqueio de aplicativos Root Firewall 83
Figura 66 - Menu da ferramenta Root Firewall 83
Figura 67 - Tela Inicial Bluetooth Firewall 85
Figura 68 - Menu da ferramenta Bluetooth Firewall 85
Figura 69 - Menu ferramenta Bluetooth Firewall 85
Figura 70 - Modo de Bloqueio por aplicativo Avast! 87
Figura 71 - Regra customizada Avast! 87
Figura 72 - Imagem da tela da ferramenta VPN One Click 88
Figura 73 - Inicio Hideman VPN 89
Figura 74 - Configurações da ferramenta Hideman VPN 89
Figura 75 - Seleção de país para conexão 90
Figura 76 - Página Viatun 91

LISTA DE ABREVIATURAS E SIGLAS

AES – *Advanced Encryption Standart*

API – *Application Programming Interface*

APP – *Application*

ARP – Address *Resolution Protocol*

BSD – *Berkeley Software Distribution*

BYOD – *Bring Your Own Device*

CSS – *Cascading Style Sheet*

DEX – *Dalvik Executable*

DOM – *Document Object Model*

GSM – *Global System for Mobile Communications*

GUI – *Graphical User Interface*

HTTPS – *HyperText Transfer Protocol Secure*

IDC – *International Data Corporation*

IP – *Internet Protocol*

MAC – *Media Access Control*

MIT – *Instituto de Tecnologia de Massachusetts*

MMS – *Multimedia Messaging Service*

NIST – *National Institute of Standards and Technology*

OHA – *Open Handset Alliance*

PIN – *Personal Identification Number*

ROM – *Read Only Memory*

SDK – *Software Development Kit*

SHA – *Secure Hash Algorithm*

SMS – *Safety Management System*

TCP – *Transmission Control Protocol*

UID – *Unique Identification Number*

VPN – *Virtual Private Network*

WPA – *Wifi Protected Access*

SUMÁRIO

1 INTRODUÇÃO 17
1.1 MOTIVAÇÃO 18
1.2 OBJETIVOS 20
1.2.1 Objetivos Gerais 20
1.2.2 Objetivos específicos 20
2 PLATAFORMA ANDROID 21
2.1 CONCEITOS DE SEGURANÇA EM REDES 25
2.1.1 Segurança na Internet 26
2.1.2 Principais Ameaças 26
2.1.3 Principais Ataques 28
2.1.4 Técnicas de Segurança 31
2.2 MODELO DE SEGURANÇA ANDROID 36
2.2.1 Linux Security 36
2.2.2 The Application Sandbox 36
2.2.3 Partição do Sistema e modo de segurança 37
2.2.4 Sistema de Permissões de Arquivos 37
2.2.5 Sistema de Criptografia de Arquivos 37
2.2.6 Proteção por senha 38
2.2.7 Utilização do Root no Dispositivo 38
2.2.8 Device Administration 39
2.2.9 Segurança das Aplicações 39
2.2.10 Permissões concedidas pelo usuário 40
3 VULNERABILIDADES DA PLATAFORMA ANDROID 41
4 CRITÉRIOS PARA OS TESTES DE FERRAMENTAS DE SEGURANÇA 44
4.1 CRITÉRIOS UTILIZADOS PARA OS TESTES COM ANTIVÍRUS 44
4.2 CRITÉRIOS UTILIZADOS PARA OS TESTES COM FIREWALL 45
4.3 CRITÉRIOS UTILIZADOS PARA OS TESTES COM VPN 46
5 TESTE DAS FERRAMENTAS DE SEGURANÇA ANDROID 47
5.1 METODOLOGIA ADOTADA NO TESTE DE SEGURANÇA 47
5.2 SELEÇÃO DAS FERRAMENTAS DE SEGURANÇA 48

5.2.1 Fermentas de Antivírus: ... 48
5.2.2 Ferramentas de Firewall: ... 51
5.2.3 Ferramentas de VPN ... 52
5.3 EXECUÇÃO DO TESTE DE INVASÃO ... 52
5.3.1 Ataque via Bluetooth ... 55
5.3.2 Ataque Via Rede Wi-fi ... 58
5.3.3 Ataque via Browser ... 60
5.3.4 Testes Utilizando Aplicativo ... 63
5.3.5 Teste realizado com SMS e Ligações Telefônicas ... 65
6 RESULTADOS OBTIDOS COM OS TESTES ... 66
6.1 FERRAMENTAS DE ANTIVÍRUS PARA A PLATAFORMA ANDROID 66
6.1.1 AVG Antivírus - Free ... 66
6.1.2 Lookout Security & Antivírus ... 69
6.1.3 Dr. Web Antivírus Light ... 71
6.1.4 Avast ! Mobile Security ... 73
6.1.5 Norton Mobile Security ... 75
6.1.6 Eset Mobile Security ... 77
6.2 FERRAMENTAS DE FIREWALL PARA A PLATAFORMA ANDROID . 79
6.2.1 DroidWall ... 80
6.2.2 Root Firewall ... 82
6.2.3 Bluetooth Firewall Trial ... 84
6.2.4 Avast Mobile Security & Antivirus ... 86
6.3 FERRAMENTA VPN ... 87
6.3.1 VPN One Click ... 88
6.3.2 Hideman VPN ... 89
6.3.3 Viatun 4 VPN ... 90
7 COMPARAÇÃO ENTRE AS FERRAMENTAS DE SEGURANÇA ... 92
7.1 COMPARAÇÃO ENTRE AS FERRAMENTAS DE ANTIVÍRUS ... 92
7.1.1 Comparativo das ferramentas de antivírus no critério eficiência ... 92
7.1.2 Comparativo das ferramentas de antivírus no critério desempenho ... 93
7.1.3 Comparativo das ferramentas de antivírus no critério tempo de respos... 94
7.1.4 Avaliação das melhores ferramentas de antivírus ... 95
7.2 COMPARAÇÃO ENTRE AS FERRAMENTAS DE FIREWALL ... 96

7.2.1 Comparativo das ferramentas de firewall no critério eficiência 96
7.2.2 Comparativo das ferramentas de firewall no critério desempenho 97
7.2.3 Avaliação das melhores ferramentas de firewall 98
7.3 COMPARAÇÃO ENTRE AS FERRAMENTAS DE VPN 98
7.3.1 Comparativo das ferramentas de VPN no critério desempenho 98
7.3.2 Avaliação da melhor ferramenta de VPN 99
7.4 AVALIAÇÃO TOTAL DAS FERRAMENTAS 99
8 CONSIDERAÇÕES FINAIS 100
REFERÊNCIAS 102
ANEXOS 105

1 INTRODUÇÃO

A telefonia móvel popularizou-se de forma rápida e cada vez mais acessível. Hoje em dia, é inevitável não utilizar um dispositivo móvel, principalmente um *smartphone*, que possui plataformas e várias opções de aplicativos, tornando-se um simples telefone celular em um microcomputador.

Com o aumento da utilização de dispositivos móveis, principalmente de *smartphones*, as vulnerabilidades também aumentaram, já que os dispositivos móveis estão no alvo de *crackers*, pois seus ataques antes realizados em PC's, agora ganham outra versão, porém com mesmo nome, como *Phishing*, *Trojans*, *Spyware*, *Bots*, *Root Exploits,* entre outros. Ainda existe o fator de que os usuários de dispositivos não utilizam ferramentas para melhorar a segurança do dispositivo ou não possuem conhecimento, uma vez que acreditam na total segurança do sistema.

Mas não apenas usuários comuns são alvos de atacantes, as empresas também são alvos de atacantes, que representam uma das maiores preocupações para empresas que passaram a utilizar dispositivos móveis, pois essas temem por roubo de informações importantes. Devido ao fato de um dispositivo conectar em diversas redes, aumentam as chances de contrair um vírus, *malwares* e outras ameaças presentes na interconectividade atual. O fenômeno chamado BYOD (*Bring Your Own Device*), pode facilitar a entrada destas vulnerabilidades em empresas e organizações, uma vez que funcionários levam seus dispositivos móveis para dentro de empresas e organizações, e com eles acessam dados e programas. Também empresas não possuem políticas de segurança para a utilização de dispositivos por seus funcionários, sendo que estes carregam dados importantes e às vezes até confidenciais destas empresas.

Sendo assim,

> A segurança deve ser trabalhada em todos os pontos possíveis, seguindo-se o conceito de defesa em profundidade ou em camadas. Assim, todo o elemento que manipule direta ou indiretamente a informação, incluído ela própria deve ter sua segurança trabalhada. (Silva, 2008, p.3)

Assim é aconselhável que empresas e usuários de dispositivos móveis, obtenham segurança utilizando ferramentas próprias para esta finalidade.

Considerando os fatores relatados acima, o trabalho a seguir apresenta uma pesquisa referente à Plataforma *Android* e suas vulnerabilidades, bem como as ferramentas existentes para a segurança dos dispositivos móveis com esta plataforma. Dentre os tópicos apresentados no trabalho a seguir, estão os ataques possíveis a serem utilizados em dispositivos móveis, bem como uma descrição de quem são os atacantes, também das políticas de segurança para utilização de dispositivos móveis e como funciona o modelo de segurança do dispositivo *Android* e algumas das vulnerabilidades presentes nesta plataforma. Após são apresentadas as ferramentas de segurança escolhidas para a realização dos testes, os critérios utilizados, os testes de invasão utilizando o Sistema Operacional Backtrack, os resultados obtidos e por último uma comparação entre as ferramentas, demonstrando os resultados em forma de gráfico permitindo visualizar as ferramentas com melhores ou piores resultados.

1.1 MOTIVAÇÃO

A demanda por dispositivos móveis aumentou significativamente nos últimos anos, sendo que entre seus utilizadores estão desde usuários leigos até usuários avançados, e estes possuem em comum a necessidade de utilizar serviços que forneçam alguma interatividade ou que acrescentem algum recurso de portabilidade às suas atividades cotidianas.

O uso de dispositivos móveis está refletindo no estilo de vida das pessoas, já que este traz facilidades e serviços antes só disponibilizadas em PC's. Além dos serviços, os dispositivos móveis também herdaram dos PC's algumas vulnerabilidades. Segundo o relatório da McAfee (2012), os dispositivos móveis não estão livres de problemas de segurança, uma vez que seus idealizadores podem lançar no mercado, dispositivos móveis vulneráveis já que estes não são suficientemente testados quanto a sua segurança.

Assim,

> Os dispositivos móveis têm-se tornado parte integrante da sociedade e, para alguns, uma ferramenta essencial. No entanto, a funcionalidade e concepções complexas destes dispositivos introduziram vulnerabilidades adicionais. Essas vulnerabilidades, juntamente com a quota de mercado em expansão tornam a tecnologia móvel um alvo atraentemente viável e gratificante para aqueles interessados em explorá-las. (SACHSE, 2010. p.6).

Com a utilização de dispositivos móveis, muitas vezes seus usuários não possuem o conhecimento suficiente para ficarem protegidos contra ameaças, beneficiando pessoas mal-intencionadas, uma vez que estas têm como foco explorá-los com roubo de senhas, envio de mensagens sem o conhecimento do usuário dentre outros ataques.

Segundo o relatório do ESET, entre os dispositivos móveis, a Plataforma *Android* é a mais utilizada e consequentemente a com maior número de ataques. Segundo AVTEST (2012), no *Android Market* existem cerca de 450.000 aplicativos disponíveis para seus usuários, porém com o crescimento deste mercado, também cresce ameaças de ataques mal-intencionados. Dentre as ameaças encontradas na Plataforma *Android*, encontram-se: *Phishing*, *Trojans*, *Spyware*, *Bots*, *Root Exploits*, fraude via *SMS*, *Premium Dialers* e falsos instaladores.

Para uma maior segurança em relação aos ataques de dispositivos móveis, é necessário a utilização de estratégias para combatê-las, como a utilização de antivírus, *firewall*, e outras formas de segurança existentes para dispositivos móveis.

É frequente o uso de *Smartphones* com a Plataforma *Android* em ambientes corporativos, sendo que estes utilizam a rede *WiFi* da empresa para se interconectar através de diversas ferramentas, com isso as vulnerabilidades presentes nestes dispositivos se torna uma ameaça para a própria empresa. Segundo Dr. Jeffrey Voas, membro da *Institute of Electrical and Electronics Engineers* (2012), as empresas deverão ser os principais alvos de *hackers* e *crackers* em 2012, sendo que em 2011 os ataques foram frequentes em redes sociais. Conforme o ESET (2013), usuários de dispositivos móveis em empresas, podem acessar as redes internas da empresa, obtendo dados importantes, sendo que uma vez este dispositivo estando infectado, possibilita atacantes em obter dados importantes da empresa. Além da utilização de redes *Wifi* para realizar ataques, os *hackers* e *crackers* também utilizam outros meios como o *SMS* e *e-mail*. Apesar do crescimento de ataques, a segurança presente nos dispositivos móveis nem sempre o acompanha, faltando conhecimento tanto de empresas quanto para usuários comuns de como se protegerem dos atacantes de dispositivos móveis.

A motivação por esta área e assunto, se dá pelo aumento na utilização de dispositivos móveis pela população e com isso o aumento de ataques. Também existe a falta de conhecimento e cuidados por parte dos usuários destes dispositivos que não possuem noção de todos os problemas que a falta de segurança pode trazer a eles,

obrigando ao profissional de T.I. a encontrar formas de aumentar a segurança para estes usuários.

Portanto, o foco do projeto se dará em encontrar os principais problemas existentes na segurança em dispositivos móveis e apontar como as soluções existentes os protegem.

1.2 OBJETIVOS

1.2.1 Objetivos Gerais

Identificar as principais vulnerabilidades de segurança na Plataforma *Android* e como ferramentas de segurança disponíveis para a Plataforma os protegem.

1.2.2 Objetivos específicos

- Identificar as deficiências de segurança existentes na Plataforma *Android*;
- Destacar as ameaças à segurança existentes em redes;
- Definir principais ameaças à Plataforma *Android*;
- Testar e avaliar ferramentas de segurança para a Plataforma *Android;*

2 PLATAFORMA *ANDROID*

A utilização de dispositivos móveis é uma constante nos dias atuais, sendo que estes possuem arquiteturas e plataformas diferentes. Conforme RODRIGUES *apud* Morimoto (2009),

> Para a utilização de todo o potencial proporcionado pelos smartphones existe a necessidade de sistemas operacionais, que neste caso também são chamados de plataforma já que com raras exceções podem ser substituídos e são programados com adaptações específicas para cada modelo de celular.

Segundo a IDC Brasil[1], o mercado de smartphones em 2012 cresceu cerca de 84% em relação ao ano de 2010, segundo RODRIGUES *apud* Morimoto (2009), entre os principais sistemas operacionais existentes para *smartphones*, encontram-se *IPhoneOS*, *BlackBerry*, *Windows Mobile* e *Android*, sendo que o primeiro foi desenvolvido pela empresa *Apple* unicamente para os seus próprios smartphones, o segundo foi desenvolvido pela empresa *RIM* para os *smartphones BlackBerry*, o *Windows Mobile* desenvolvida pela empresa *Microsoft* não possuindo um *smartphone* especifico para sua utilização, e o *Android* que é uma Plataforma desenvolvida pela empresa Google, a qual obteve o sistema pré-pronto de uma empresa em 2005, a Google idealizou o Plataforma *Android* como *open-source*.

Segundo RODRIGUES *apud* Morimoto (2009), pelas funcionalidades disponíveis e diversas definições, podemos classificar os *smartphones* como dispositivos programáveis que convergem mobilidade e conectividade.

A plataforma *Android* foi a solução da *Google* juntamente com a *OHA(Open Handset Alliance)* para ocupar o mercado de *smartphones*. Segundo Lecheta,

> a Open Handset Alliance (OHA) é um grupo formado por gigantes do Mercado de telefonia de celulares liderados pelo Google. Entre alguns integrantes do grupo estão nomes consagrados como a HTC, LG, Motorola, Samsung, Sony Ericsson, Toshiba, Sprint Next, China Mobile, ASUS, Intel, Garmin e muito mais. (2010, p.21)

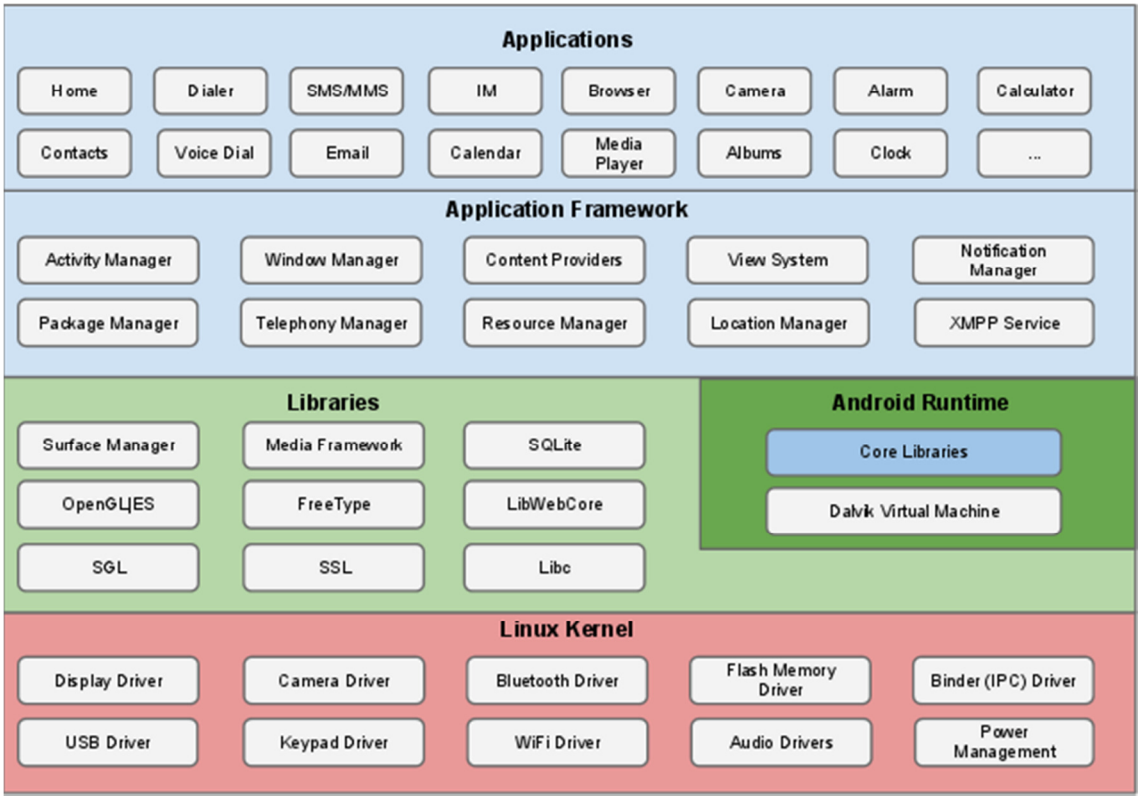
Quando foi anunciada, em 2008, a nova plataforma para dispositivos móveis *Android* causou um grande impacto, tornando-se uma concorrente para o *IPhoneOS* que iniciou suas atividades em 2007, já possuindo a maioria dos utilizadores de

[1] http://www.idclatin.com/news.asp?ctr=bra&year=2012&id_release=2213

smartphones. Segundo o site oficial do *Android*[2], "[...] O objetivo principal do *Android* é criar uma plataforma de software aberto disponível para as operadoras OEM (Original Equipament Manufacturer) e desenvolvedores para transformar suas ideias inovadoras em realidade [...]".

A plataforma *Android* foi concebida para ser completamente open-source, no qual os desenvolvedores podem criar aplicativos utilizando as diversas funcionalidades presentes na plataforma. A arquitetura do *Android* é composta por camadas, sendo elas, o *Kernel GNU Linux*, Bibliotecas, Framework de Aplicação e Aplicação, como é demonstrado na figura 1.

Figura 1 - Arquitetura do Android



Fonte: http://source.android.com/tech/security/index.html

Cada uma das camadas funciona da seguinte forma:

- *Kernel Linux*: A versão do *Kernel Linux* presente na plataforma *Android* é a 2.6, que é o responsável pelos principais serviços do sistema, como a segurança, gestão de memória, gestão de processos, pilha de protocolos de rede e modelo de drivers. Porém, "apesar de ter sido construído com

[2] http://source.android.com/source/index.html

base no *Linux*, não é um *Linux*, não possui *Windowing system* nativo (componente de GUI[3]) não suporta glibc e não possui algum dos conjuntos de padrões apresentados em algumas distribuições *Linux*.". (Pereira; Silva, 2009, p.4)

- Bibliotecas: Possui um conjunto de bibliotecas de C/C++, "bibliotecas de multimídia, visualização de camadas 2D e 3D, funções para navegadores *web*, funções para gráficos, funções de aceleração de hardware, renderização 3D, fontes *bitmap* e vetorizadas e funções de acesso ao banco *SQLite*." (Pereira; Silva, 2009, p.7). Estas bibliotecas podem ser acessadas por desenvolvedores através de *frameworks*, as principais bibliotecas são: *Freetype* - renderização de fontes e *bitmaps*, *System* C *Library*- Derivação da biblioteca C do BSD, e biblioteca C *Bionic*, desenvolvida especialmente para o *Android*, *WebKit*- Renderizador de páginas para navegadores com suporte para *javascript*, *CSS*, *DOM* e *Ajax*, *SQLite*- Banco de Dados desenvolvida em C.
- *Android Runtime*: Utiliza uma instância de máquina virtual *Dalvik* desenvolvida para dispositivos móveis, onde cada aplicação executa seu próprio processo. "A *Dalvik* executa arquivo .DEX(*Dalvik Executable*), que são classes *Java* convertidas para a máquina virtual através da ferramenta DX, distribuída juntamente com o SDK[4] do *Android*." (Pereira; Silva, 2009, p.8)
- *Framework* de Aplicação: Camada onde se encontram todas as aplicações e seus recursos, entre alguns dos elementos presentes na camada estão: *Gerenciador de atividade*, responsável por iniciar e terminar uma *atividade*; *Package Manager*, responsável por ler os pacotes de arquivos do *Android*, sendo utilizado pelo *Gerenciador de atividade*, definindo quais pacotes estão sendo utilizados bem como sua capacidade; *Window Manager*, gerenciador de janelas; *Content Providers*, permite a comunicação entre aparelhos e a troca de informações entre aplicativos; *View System*, determina a aparência de determinada aplicação, como botões, *layouts*, etc. Com esta arquitetura, é

[3] GUI-Interface gráfica com o usuário.
[4] Android SDK-Kit de desenvolvimento destinado a programadores de aplicativos para a plataforma Android.

possível a reutilização de código por desenvolvedores, sendo que "qualquer aplicação pode publicar as suas capacidades e qualquer outra aplicação candidata pode, então, fazer uso dessas capacidades." (Pereira; Silva, 2009, p.7)

- Aplicação: É a camada mais alta, encontram-se nesta camada os principais aplicativos utilizados por usuários, como *email*, navegador, mapas, etc. Todos os aplicativos presentes nesta camada utilizam a linguagem *Java*.

3 CONCEITOS DE SEGURANÇA EM REDES

Como a utilização da Plataforma *Android* em dispositivos móveis é semelhante ao uso de sistemas operacionais específicos para *PC's*, existem conceitos de segurança que também são semelhantes aos utilizados para computadores em rede, sendo que para haver uma verdadeira segurança em redes ou na *internet* é necessário seguir alguns dos elementos destes conceitos. Segundo Martins *apud* Carvalho(2005), "os usuários de serviços de redes ou *internet*, devem estar adaptados às regras e padrões presentes nas políticas de segurança, sendo que assim seus dados permanecerão íntegros, confiáveis e disponibilizáveis". Para a criação de políticas de segurança, são necessários passos a serem seguidos, conforme Carvalho (2005, p.56), "[...] estes passos são a definição do escopo da política de segurança e a identificação de contra quem a informação está sendo protegida".

É importante a compreensão de como definir as políticas de segurança, uma vez que é com estas políticas que uma rede ou usuários estarão realmente seguros de ataques. Com uma definição mal sucedida de políticas de segurança, toda a rede e seus usuários estarão suscetíveis a diversos ataques, com consequências muitas vezes irreversíveis. A seguir são listados quais os principais itens que uma política de segurança deve seguir para ser bem sucedida:

- Definição do Escopo: Para definir o escopo da política de segurança, é necessário possuir conhecimento sobre o que deve ser protegido, como dados presentes no banco de dados, servidores, etc.;
- Identificação de ameaças: Para a identificação de ameaças, devem-se considerar segundo Carvalho (2005, p.57) os seguintes itens: Acesso não autorizado a recursos ou informações; Revelação não autorizada de informações; *Bugs* de sistemas e erros de usuários.

3.1 SEGURANÇA NA INTERNET

A utilização da internet se tornou uma constante nos dias atuais e isto se deve principalmente ao uso de redes sociais, às compras online e a grande facilidade de encontrar qualquer conteúdo via *web* pelos usuários. Segundo Cheswick (2005, p.87), existem ao menos quatro problemas em segurança na *web* que são necessários resolver, que são: riscos para o cliente, proteção de dados durante a transmissão, riscos diretos do servidor ao executar *software* de *web* e outros caminhos de acessar um determinado *host*.

Para dispositivos móveis, a *internet* é uma das principais ferramentas responsáveis pela sua expansão. A facilidade de conexão em redes *wi-fi* em diversos lugares públicos como *shoppings*, aeroportos, rodoviárias, dentre outros, torna ainda mais atrativa a utilização da *internet* por meio de *smartphones* e *tablets*, porém os usuários não possuem conhecimento de que justamente esta rede pode estar sendo alvo de pessoas mal-intencionadas, que podem obter senhas, *logins* ou realizar qualquer outro ataque a estes dispositivos. Assim, para uma maior segurança para empresas, órgãos públicos ou simples usuários, são necessários a utilização de políticas de segurança que corrijam problemas como os citados anteriormente.

3.2 PRINCIPAIS AMEAÇAS

Para proporcionar uma segurança eficiente, é necessário obter conhecimento de quem são os responsáveis pelos ataques existentes. Entre os atacantes contra usuários de sistemas operacionais para dispositivos móveis estão:

- *Hackers*: são indivíduos que em uma primeira impressão dão a ideia de ameaça, porém segundo Moraz (2006), o termo *hacker* ocorreu pela primeira vez em 1950 no MIT (*Instituto de Tecnologia de Massachusetts*), sendo definido como pessoas com interesse por processamento eletrônico de dados, podendo ser caracterizado também como pessoas que modificam algo que está pronto com o intuito de aperfeiçoá-lo, normalmente um hacker encontra algum problema de segurança e contata o responsável pelo sistema. Segundo Moraz (2006), o termo *hacker* pode ser ainda denominado como *hacker White hat*, *hacker gray hat* e *hacker black hat*, sendo o primeiro conhecido como o *hacker* ético, o segundo

pode-se definir como *hacker* ético que não possui intenção de roubo, porém utiliza seus conhecimentos para infiltrar-se sem nenhum tipo de autorização por parte do responsável do sistema, e o *hacker black hat* é também conhecido como *cracker*.

- *Cracker*: É o oposto do *hacker White hat*, pois seu principal intuito é o de cometer crimes virtuais, ele normalmente possui um vasto conhecimento de informática e trabalha em conjunto com outros *crackers.* Os *crackers* também possuem classificações, como *crackers* de sistema e *crackers* de programas. Conforme Moraz,

> Crackers de sistemas são invasores de sistemas interligados em redes. As potenciais e mais agressivas ações referem-se à obtenção de acessos a informações sigilosas, à destruição destas, ou à obtenção para uso em beneficio próprio. Isso, no entanto não limita as ações dos crackers de sistemas a ambientes corporativos, crackers de sistema podem, com relativa facilidade, burlar computadores domésticos, transformando a vida do usuário comum em um verdadeiro inferno, sem que este saiba, de fato, o que pode estar ocorrendo, pois a diretiva de um cracker é, evidentemente, não ser descoberto. (2006, p.35)

Já os *crackers* de programas são indivíduos que para a utilização de programas pagos, tornam estes utilizáveis sem nenhum custo, tanto para o próprio *cracker* como para usuários comuns.

- *Phreackers*: Estes têm como intuito a invasão de telefonia, fixa, pública ou móvel, para a utilização de seus serviços sem nenhuma cobrança em valor monetário.

Os atacantes existentes contra sistemas de dispositivos móveis são os mesmos encontrados para sistemas comuns. A crescente utilização de dispositivos móveis com diversos tipos de sistemas operacionais por usuários indefesos pode ter influenciado os atacantes a direcionar sua atenção para estes dispositivos, utilizando ataques já existentes em sistemas para *desktops*, modificados para invadir especialmente as diversas plataformas para dispositivos móveis.

3.3 PRINCIPAIS ATAQUES

Assim como ter o conhecimento de quem efetua o ataque é importante também saber quais os principais ataques cometidos por estes, para proporcionar uma maior segurança em redes. Dentre eles, podemos destacar:

- *Phishing*: Segundo Moraes (2011, p.16), este tipo de ataque tem como objetivo adquirir dados de usuários, como números de CPF, RG, senhas e etc.. Para isso os atacantes utilizam Engenharia Social, se passando por empresas conhecidas, muitas das vezes oferecendo oportunidades de ganhos ou necessitando alguma alteração cadastral. Os ataques costumam acontecem por *email*, onde usuários tanto de *PC's* quanto dispositivos móveis são alvos.

- *Rootkits*: Segundo Morimoto "*rootkits* são softwares que exploram um conjunto de vulnerabilidades conhecidas para tentar obter privilégios de *root*." (2008, p. 393). Eles se instalam no sistema sem que este seja percebido por antivírus, é uma ameaça que vem crescendo em dispositivos móveis, segundo o relatório da McAfee (2011,p.23), os *Rootkits* podem ser instalados nos dispositivos móveis enquanto estes estão em lojas e fábricas, sem estar ao poder do usuário, ou seja, o usuário receberia o dispositivo já infectado.

- *Trojans*: Segundo Moraes (2011, p.5), são programas que uma vez instalados no sistema acabam possuindo acesso para execução de qualquer tarefa. Segundo a Kasperky Lab (2012)[5], o *Trojan-SMS* é o mais comum a atacar os dispositivos móveis da plataforma Android. Este tipo de ameaça é instalada no dispositivo, enviando mensagens *SMS* de custo elevado, trazendo prejuízos ao usuário.

- *Spyware*: Segundo Moraes (2011, p.8), são arquivos que se infiltram no sistema sem o usuário perceber, como diz o nome é um arquivo espião, que pode roubar dados do usuário, segundo a Kaspersky Lab (2012)[6], este tipo de ameaça é a terceira mais frequente em dispositivos móveis.

[5] http://brazil.kaspersky.com/sobre-a-kaspersky/centro-de-imprensa/blog-da-kaspersky/trojan-sms-br
[6] http://www.kaspersky.com/pt/news?id=207576091

- *Bootnets*: Segundo Cheswick, Bellovin e Rubin (2005), são programas planejados para realizar uma determinada tarefa, que se propaga através de rede, normalmente é utilizado para envio de *spams* e propaganda, este tipo de ataque pode ser implementado tanto por computador quanto por dispositivo móvel. Segundo a McAfee (2011), entre os ataques que podem ser efetuados por redes de *bots* para dispositivos móveis, estão desde o aumento da largura de banda fora do protocolo TCP/IP, até o ataque em operadoras, provocando negação de serviços, também este tipo de ataque pode chegar a usuários comuns com envios maciços de mensagens SMS ou chamadas de voz.

- Furto de Senhas: Segundo Cheswick, Bellovin e Rubin (2005), invasões de sistemas são facilmente realizadas devido a falhas presentes no sistema de senhas, entre as causas de furto de senhas, a mais comum é a de o usuário utilizar uma senha ruim, que pode vir a ser facilmente adivinhada pelo atacante. Existem duas formas de adivinhações por invasores: uma em que são efetuados *logins* em sistemas utilizando nomes de usuários conhecidos e supostas senhas; na segunda forma, os invasores comparam adivinhações contra arquivos de senhas roubadas de um sistema, caso um sistema já tenha sido invadido, os atacantes tentarão invadir outros sistemas, já que usuários normalmente repetem a senha utilizada no sistema anterior, sendo os chamados ataques de dicionários que costumam ser muito bem sucedidos. Nos dispositivos móveis, a utilização de senhas e *logins* se dá para diversos tipos de serviços, como para acesso a bancos, para a verificação de contas, acesso a *emails*, também é muito utilizado para o bloqueio de segurança do *smartphone*. Segundo o relatório da McAfee (2011, p.20), os usuários de *smartphones* utilizam senhas de bloqueio do dispositivo ou PIN's de fácil memorização, como 0000, 1234, etc. Este é um exemplo de como usuários não possuem conhecimento da utilização de senhas mais robustas para uma maior proteção de seus dispositivos móveis.

- *Bugs* e *Backdoors*: Segundo Cheswick, Bellovin e Rubin (2005), esses ataques se beneficiam de erros presentes em sistemas deixados pelos desenvolvedores do sistema, ou seja, deixa uma porta aberta para que o ataque ocorra. Esta vulnerabilidade está presente também nos dispositivos móveis, uma vez que a maioria dos aplicativos são criados por desenvolvedores que disponibilizam suas

criações para *download*, sendo que muitas vezes em locais não muito seguros. Assim estes aplicativos podem possuir algum tipo de *bug* ou código maliciosos, e as empresas desenvolvedoras de plataformas para dispositivos móveis poderão não ter conhecimento destes aplicativos já que estes não se encontram em locais oficiais de *download* destes sistemas.

- Ataques exponenciais – Vírus e *Worms*: Segundo Moraes (2011, p.4,7), são tipos de ataques que se multiplicam rapidamente através da infecção de outros programas, quando estes programas não necessitam de ajuda para se propagar, são considerados *worms*, já se o programa necessita de outros para se propagar, são considerados vírus. Os vírus e *worms* podem também ser encontrados em dispositivos móveis, assim como em *PC's* eles podem se propagar através de conteúdos de procedência duvidosa, como *emails* infectados, sites não confiáveis e, especialmente em plataformas de dispositivos móveis, aplicativos contaminados com vírus ou *worms*.

- *Exploit*: Segundo Giavaroto e Santos (2013, p.127), são códigos de programas desenvolvidos com a intenção de explorar alguma vulnerabilidade de sistemas. Um *exploit* utiliza *payload*, que é uma parte do código que abre uma comunicação entre o atacante e o atacado, podendo obter dados importantes do sistema.

- *ARP Poison*: É um tipo de ataque utilizado em redes, onde os pacotes enviados por toda a rede são interceptados pelo atacante, uma vez que todo o tráfego da rede é redirecionado para o computador do atacante, segundo Morimoto (2008, p. 437), "o micro do atacante envia pacotes com respostas forjadas para requisição ARP de outros micros da rede. [...] o ARP é utilizado para descobrir os endereços MAC dos demais micros da rede.", os dispositivos móveis também estão suscetíveis a estes tipos de ataques uma vez que estes habitualmente utilizam acesso a redes para realizar alguma tarefa.

Conhecendo atacantes e seus principais ataques, é possível considerar que tanto os sistemas *desktops* quanto sistemas desenvolvidos para dispositivos móveis,

encontram-se em risco, ainda mais com a utilização de políticas de segurança mal planejadas, essas vulnerabilidades tornam-se ainda mais visíveis a seus atacantes.

Porém com a utilização correta dos mecanismos de segurança disponíveis para estas tecnologias, os ataques podem ser minimizados ou totalmente extintos de redes e sistemas operacionais.

3.4 TÉCNICAS DE SEGURANÇA

Para que os ataques vistos anteriormente possam ser evitados ou descartados é necessário haver, tanto para corporações como para usuários comuns, conhecimento de segurança para poder manter seu sistema e suas informações intactas.

Na concepção de segurança para dispositivos móveis, alguns itens são considerados relevantes para sua aquisição, segundo Myerson (2011), ”dentre estes itens estão: a verificação de regras de firewall configuráveis, a eficiência da criptografia, e o número máximo de tentativas para senha [...]”. Também podem ser consideradas algumas medidas de segurança que são utilizadas em *desktops*, como: sistema de detecção de intrusão, redes privadas virtuais e autenticação. A seguir são descritas algumas dos itens listados anteriormente.

3.4.1 Sistemas de *Firewall*

Segundo Carvalho (2005, p.15), ”um sistema de firewall pode ser definido como dispositivo que combina software e hardware para segmentar e controlar acesso entre redes de computadores distintas”. Para Morimoto (2008, p.396),

> “o firewall trabalha verificando os endereços de origem e de destino dos pacotes, portas a que são destinados e o status das conexões. Ele não é destinado a verificar o conteúdo dos pacotes e por isso pouco pode fazer com relação a vírus, trojans, Phishing e outros ataques similares. Para eles temos os antivírus, que trabalham verificando o conteúdo dos arquivos acessados.”

Um *firewall* permite ao usuário, ou ao responsável pelo serviço de rede, gerenciar o que pode ou não passar por esta "parede de fogo". Com a utilização do *firewall*, a segurança nos sistemas operacionais pode ser aumentada significativamente.

3.4.2 Antivírus

Assim como visto anteriormente para um sistema possuir maior segurança é necessária a utilização de ferramenta para detecção de intrusos na plataforma, uma vez que o *firewall* não verifica o conteúdo de pacotes recebidos pelo usuário. Conforme Moraes (2011, p.36), "antivírus é um software utilizado para inspecionar o conteúdo de arquivos e tentar localizar códigos que também são chamados de assinaturas e comparar com o seu banco de dados de vacinas para interromper uma possível infecção do sistema [...]".

Para que um antivírus possa funcionar de forma eficiente é necessário utilizar antivírus de empresas conceituadas e conhecidas, uma vez que ao utilizar *softwares* de empresas desconhecidas se corre o risco em tornar vulnerável a plataforma. Também é necessária a atualização permanente do *software* antivírus, já que novos vírus são criados periodicamente pelos invasores, caso o antivírus não esteja atualizado e um novo vírus atacar o sistema, ele não será capaz de detectá-lo, deixando a plataforma vulnerável.

Para os dispositivos móveis são inúmeras as empresas que desenvolvem aplicativos de antivírus. Segundo o relatório de testes contra vírus da empresa AVTest(2012)[7], conforme Anexo A, das empresas que desenvolvem aplicativos de antivírus para dispositivos com *Android*, apenas 10 empresas ficaram entre as mais eficazes na detecção de vírus, ficando entre 90% ou mais em sua eficácia. As empresas que ficaram entre 90% e 65% foram consideradas com bons produtos, os produtos que tiveram eficiência entre 65% e 40% foram considerados eficazes apenas para alguns tipos de vírus e os produtos que detectaram abaixo de 40% foram consideradas muito ineficazes quanto ao combate a vírus existentes. Alguns aplicativos desta categoria podem ser encontrados na loja de aplicativos do sistema *Android-Play Store*, o teste ainda indica que os aplicativos de antivírus não são perfeitos, com alguns problemas ainda a serem resolvidos por seus fabricantes, como armazenamento de relatórios de testes.

Apesar de alguns antivírus não serem completamente eficazes em seu tratamento de vírus, ainda é considerada uma ferramenta importante para a detecção e remoção de vírus presente na plataforma, apenas é necessário obter alguns critérios para sua utilização, como procedência e eficácia.

[7] http://www.av-test.org/fileadmin/pdf/avtest_2012-02_android_anti-malware_report_english.pdf

3.4.3 Criptografia

A criptografia tem um papel fundamental para a segurança em redes, na transmissão de dados e na segurança de arquivos. A criptografia tem como princípio a segurança de dados, permitindo o ciframento dos dados, sendo a cifra constituída de um ou vários algoritmos, que tornam difícil o deciframento da chave por pessoas mal-intencionadas. As chaves de criptografia, segundo Alecrim, "Trata-se de um conjunto de bits baseado em um determinado algoritmo capaz de codificar e de decodificar informações." (2009). As chaves podem possuir tamanhos diferentes, como de 64 bits, 128 bits e assim por diante, quanto maior o tamanho da chave mais difícil será para decifra-la.

Para que o sistema de criptografia funcione, é necessário haver pelo menos uma chave para o ciframento e deciframento. Na figura 2 é possível visualizar o sistema de criptografia:

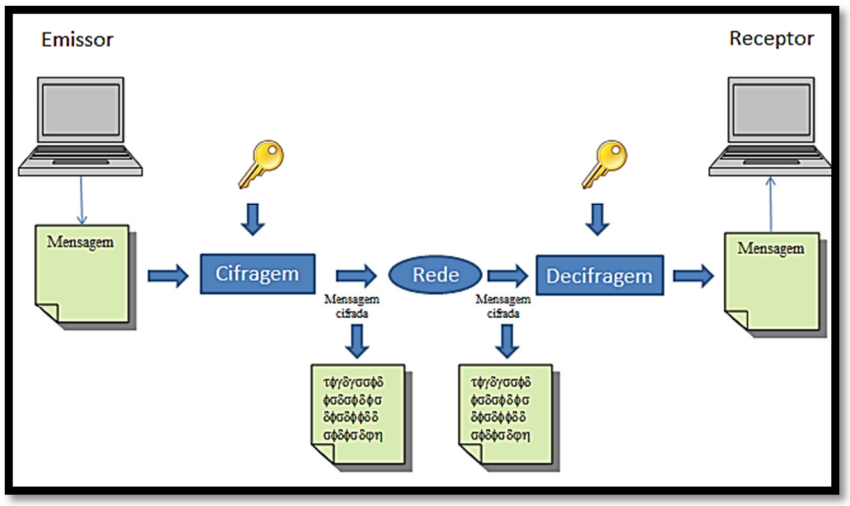
Figura 2 - Sistema de criptografia

Fonte: da Autora

Entre os tipos de criptografia existem dois, a criptografia simétrica e a criptografia assimétrica:

- Criptografia Simétrica: O sistema da criptografia simétrica define que a cifragem e decifragem de uma mensagem ou dado é feita apenas por uma chave, onde o emissor envia uma mensagem, o algoritmo de criptografia cifra a mensagem a ser enviada. O receptor recebe a mensagem criptografada e decriptografada com a mesma chave que foi criada no envio da mensagem, utilizando o mesmo algoritmo. Na figura 3 é possível visualizar o funcionamento do sistema de criptografia simétrica:

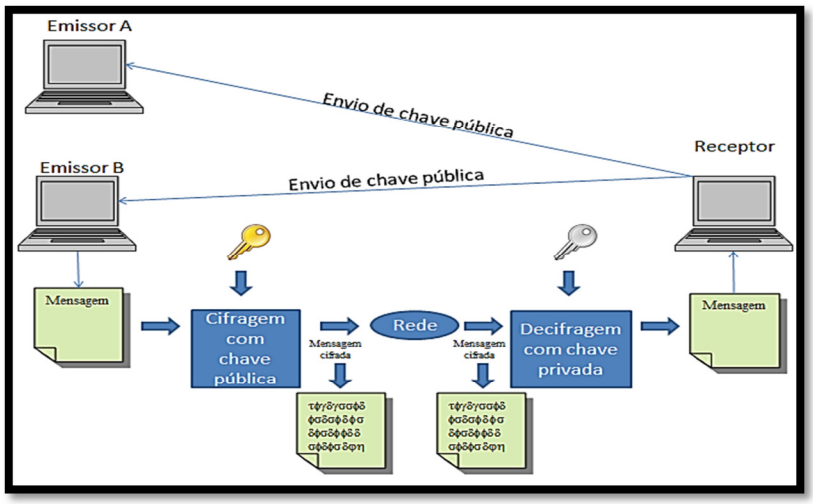
Figura 3 - Sistema de Criptografia Simétrica



Fonte: da Autora

- Criptografia Assimétrica: O sistema de criptografia assimétrica utiliza duas chaves, uma para a cifragem e outra para decifragem de mensagens. Conforme Carvalho (2005, p.40), "os sistemas de chaves públicas partem do principio de que cada pessoa interessada em transmitir ou receber informações criptografadas possua um par de chaves, uma chave chamada de chave pública e a outra chave privada". Ou seja, quando um emissor necessitar enviar alguma mensagem, ele irá utilizar uma chave pública, sendo que essa chave será disponibilizada a todos que necessitarem transmitir alguma mensagem para o receptor, assim somente o receptor terá acesso à chave privada, pois é com ela que as mensagem cifradas poderão ser decifradas. Na figura 4 é possível visualizar na imagem um sistema de criptografia Assimétrica:

Figura 4 - Sistema de Criptografia Assimétrica



Fonte: da Autora

3.4.4 Criptografia de Arquivos na Plataforma *Android*:

Na Plataforma *Android*, o sistema de criptografia de arquivos pode ser utilizado por programas criados por desenvolvedores, o sistema de criptografia de arquivos é disponibilizado pela própria plataforma *Android*, apenas a partir da versão 3.0, para as versões anteriores o sistema de criptografia é utilizado apenas para senha de segurança do dispositivo.

Para o desenvolvimento de aplicativos para a Plataforma *Android*[8], é disponibilizado pacotes para a implementação de aplicações com criptografia, que podem ser simétrica ou assimétrica. Sendo assim, para uma maior garantia de segurança das transações entre emissor e receptor, via dispositivos móveis, a utilização de criptografia é recomendada.

3.4.5 Redes Privadas Virtuais

Mais conhecidas como *VPN's*, as redes virtuais tem como fundamento a utilização de redes públicas para a sua implementação, onde dados podem ser trocados entre *hosts* que utilizem a mesma rede de forma segura por meio de criptografia e tunelamento. Segundo Carvalho (2005, p.47), "a criptografia e o tunelamento são conceitos que fundamentam a *VPN*, pois o primeiro garante a autenticidade, a confidencialidade e a integridade das conexões e o segundo permite o tráfego das informações pela rede pública". A utilização de *VPN* em dispositivos móveis pode ser considerado um ponto importante para a segurança de empresas, que necessitam trocar informações por meio de rede.

Para uma maior segurança tanto para empresas quanto para usuários, a utilização das ferramentas citadas acima é útil, pois quando bem implementadas, elas podem assegurar a segurança quase completa contra invasores ou pessoas mal-intencionadas.

[8] http://developer.android.com/reference/javax/crypto/package-summary.html

4 MODELO DE SEGURANÇA *ANDROID*

Na concepção da Plataforma *Android*, a equipe responsável pelo desenvolvimento do sistema, identificou que poderia haver possíveis vulnerabilidades presentes no sistema. Para resolver esta questão foi criado um programa de segurança, sendo os componentes presentes neste sistema: *Design Review*, Teste de Invasão, Revisão de Código, *Open Source*, Revisão da Comunidade e Respostas a Incidentes. A seguir serão descritos alguns dos recursos de segurança presentes no modelo de segurança do *Android*.

4.1 LINUX SECURITY

Para a criação da Plataforma *Android* foi utilizado o *Kernel* do *Linux* como visto anteriormente, com isso os desenvolvedores aproveitaram a segurança existente neste *kernel* para melhorar a segurança do sistema, como modelo baseado em permissão de usuário e o isolamento de processos.

4.2 THE APPLICATION SANDBOX

Para cada aplicativo instalado no dispositivo *Android*, um novo UID (*User ID*) é criado, e o APP (Aplicativo) instalado é executado sobre este UID, e para todos os dados criados por este APP é utilizado o mesmo UID criado para este aplicativo, assim as aplicações são executadas de forma isolada. Além dos aplicativos criados pelos desenvolvedores funcionarem com este modelo de execução, todos os componentes presentes na arquitetura *Android* acima do *kernel* também o utilizam, porém com todo o cuidado idealizado pelo *Android* ainda é possível ocorrerem certas vulnerabilidades na criação de aplicativos. Segundo definição do próprio sistema *Android*[9], assim como "todos os recursos de segurança, a proteção do *Application Sandbox* não é inquebrável.".

[9] http://source.android.com/tech/security/index.html#android-security-overview

4.3 PARTIÇÃO DO SISTEMA E MODO DE SEGURANÇA

A partição do sistema é composta pelo *kernel* do sistema, as bibliotecas do sistema, *android runtime, application framework* e *applications*, ou seja, todas as camadas que compõem a plataforma. A permissão da partição do sistema é apenas de leitura, ao inicializar o dispositivo móvel em modo seguro, somente serão inicializados ao sistema os aplicativos presentes na partição do sistema, ou seja, apenas os aplicativos comuns do sistema, sendo que aplicativos instalados não pertencentes ao sistema não serão inicializados. O modo de segurança pode ser utilizado no caso de algum aplicativo estiver agindo de forma inadequada no sistema, como causando alguma pane ou qualquer outro transtorno para o usuário.

4.4 SISTEMA DE PERMISSÕES DE ARQUIVOS

O sistema de permissões de arquivos do *Android* define que uma aplicação é sua própria *owner* (proprietária), assim apenas a própria aplicação terá permissão para escrita e leitura em seus arquivos, outro aplicativo somente poderá ter acesso a arquivos de outro aplicativo se este permitir que isso seja possível.

4.5 SISTEMA DE CRIPTOGRAFIA DE ARQUIVOS

O sistema de criptografia de arquivos completo é oferecido a partir da versão 3.0 da Plataforma. Segundo Cibrão e Gonçalves (2011/2012, p.9),

> A criptografia é feita ao nível do kernel, usando a implementação do dm-crypt(um subsistema do kernel linux para encriptação transparente do disco) de AES128 com CBS(Cipher Block Chaining) e ESSIV:SHA256 (Encrypted Salt-Sector Initialization Vector, usando SHA256 como função de hash).”

A chave mestra da criptografia é protegida por AES 128 utilizando uma chave derivada da senha do usuário, para aumentar a complexidade da criptografia, a senha é combinada com um salt aleatório[10] (utilizada para a proteção de senhas unix) com função SHA1. O sistema utiliza regras de complexidade de senha, para aumentar a proteção contra os ataques de adivinhação de senha.

[10] Salt: Técnica utilizada para proteção de senhas, onde a própria senha é utilizada e adicionada á um algoritmo de criptografia, onde está é gerada aleatoriamente a cada ciframento da senha. Disponível em:< http://info.abril.com.br/edicoes/252/arquivos/6236_1.shl>

4.6 PROTEÇÃO POR SENHA

A Plataforma *Android*, fornece ao usuário a possibilidade de configurar senha para bloqueio do dispositivo, assim caso um individuo tente utilizar o dispositivo móvel de um usuário será necessário inserir senha para que possa ocorrer esta utilização.

4.7 UTILIZAÇÃO DO *ROOT* NO DISPOSITIVO

O *Android* utiliza o superusuário *root* para o gerenciamento do *kernel* e um pequeno conjunto de aplicativos, por padrão o *root* nos dispositivos com a plataforma *Android* são bloqueados e inacessíveis para usuários, porém é possível realizar o desbloqueio do superusuário root, utilizando técnicas e programas, este desbloqueio pode tornar a plataforma vulnerável a ameaças, podendo qualquer recurso gerenciado pelo root ser modificado.

Uma vez possuindo permissão de *root*, é possível realizar operações em nível de *kernel*. Conforme Cibrão e Gonçalves (2011/2012, p. 26) afirmam,

> [...]apesar de fazer rooting de um dispositivo ser normalmente considerado uma operação benéfica[...], abre também um buraco na sua segurança, permitindo a aplicações maliciosas controle sobre o sistema, pelo que o utilizador deve ter completa noção dos riscos associados.

Assim, antes de realizar o desbloqueio do superusuário é necessário ter em mente que caso pessoas mal-intencionadas tiverem acesso ao dispositivo deste usuário, podem prejudica-lo de inúmeras formas. Além do desbloqueio do root, também é possível a utilização das permissões concedidas a aplicações serem utilizadas com má fé. Segundo Cibrão e Gonçalves (2011/2012, p.25), "[...] nada impede uma aplicação de requisitar permissões e nada impede o utilizador de aprová-las; no entanto, nada garante que a aplicação as use só para as funcionalidades publicitadas". Assim uma aplicação poderá se beneficiar das permissões requisitadas, e utilizar em beneficio de pessoas mal-intencionadas, cometendo todos os ataques possíveis.

4.8 DEVICE ADMINISTRATION

Para aplicações empresariais, o *Android* disponibiliza uma *API* para a administração de recursos em nível do sistema. Ela permite criar aplicativos para corporações, proporcionando ao profissional de TI a possibilidade de criar políticas de segurança específica para dispositivos móveis com a Plataforma *Android*, onde os profissionais de TI necessitem utilizar de aplicações para aumentar a segurança no ambiente empresarial, tendo controle sobre os dispositivos de funcionários da empresa.

4.9 SEGURANÇA DAS APLICAÇÕES

As aplicações do *Android* são desenvolvidas normalmente em código *Java*, estas são executadas na *Dalvik virtual machine*[11] ou podem ser desenvolvidas pelo próprio código nativo do *Android*. As aplicações possuem extensão *Apk*, e é através desta extensão que as aplicações são instaladas. Os principais itens utilizados para a criação de aplicações são: *AndroidManifest.xml*, *Activities*, *Service* e *BroadcastReceiver*, sendo que o arquivo *AndroidManifest.xml*, permite à alguma aplicação a permissão de utilizar alguns recursos presentes na Plataforma *Android*, como câmera, *GPS*, *Bluetooth*, etc. No momento em que a instalação é realizada, todas as permissões que o aplicativo necessita são concedidas. Na figura 5 é possível visualizar um exemplo deste arquivo, onde é requisitada pela aplicação a permissão para monitorar mensagens *SMS*:

Figura 5 - Modelo do arquivo AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.android.app.myapp" >
    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    ...
</manifest>
```

Fonte: http://developer.android.com/guide/topics/security/security.html

Existe a possibilidade de que algumas pessoas mal-intencionadas possam utilizar estas permissões que são dadas para a aplicação com a intenção de furtar dados do usuário, ou realizar alguma ação que proporcione ao dispositivo e a Plataforma *Android*

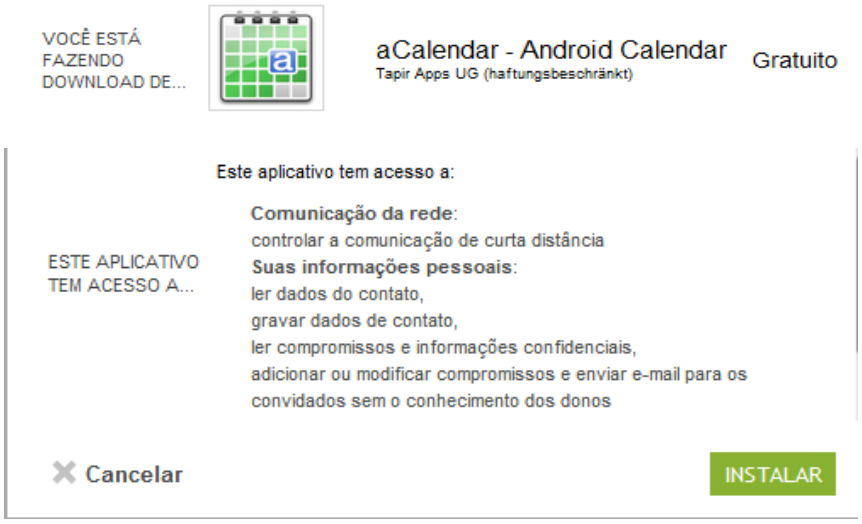
[11] *Dalvik virtual machine:* Máquina virtual criada para integrar a Plataforma *Android*, permite a execução de várias instâncias de máquinas virtuais e utilizar pouca memória do dispositivo móvel.

agir de forma incorreta, como por exemplo, enviar mensagem *SMS* sem o conhecimento do usuário.

4.10 PERMISSÕES CONCEDIDAS PELO USUÁRIO

Caso o usuário resolva utilizar alguma aplicação fornecida por terceiros, que podem não ser encontradas na loja de aplicativos da Plataforma *Android - Play*, no ato da instalação, é solicitado ao usuário se ele aceita as permissões que serão cedidas à aplicação, como é visto na figura 6.

Figura 6 - Imagem do Aplicativo aCalendar



Fonte:https://play.google.com/store/apps/details?id=org.withouthat.acalendar&feature=search_result#?t=W251bGwsMSwxLDEsIm9yZy53aXRob3V0aGF0LmFjYWxlbmRhciJd

Caso o usuário instale este aplicativo em seu dispositivo, todas as permissões solicitadas pela aplicação serão concedidas, no caso permissão de comunicação da rede e as informações pessoais do usuário.

5 VULNERABILIDADES DA PLATAFORMA *ANDROID*

O modelo de segurança da Plataforma *Android* possui algumas brechas de segurança, deixando o sistema suscetível a ataques. A seguir são descritos alguns dos ataques cometidos contra a Plataforma *Android*, bem como uma definição de como agem, em qual parte do sistema eles possuem acesso e quais danos eles causam ao sistema e ao usuário.

As lojas de Aplicativos falsas são uma das vulnerabilidades que atingem a Plataforma *Android*, visto que é possível que seus usuários utilizem lojas que não são autorizadas a disponibilizar aplicativos, facilitando a manifestação de *malwares*. Segundo o site seumicroseguro (2012)[12], dentre os transtornos que estes aplicativos causam está a permissão para leitura de mensagens SMS e MMS, saber a localização do dispositivo, acesso da internet, apagar o conteúdo armazenado do dispositivo e coletar informações de chamadas telefônicas de seu usuário.

Existem também vulnerabilidades das API's de Segurança da Plataforma *Android*. Segundo o relatório da Mcafee (2011)[13], dentre as vulnerabilidades das API's estão nas suas descrições quando são conflitantes ou inexistentes, as trocas de mensagens entre processos podem ser interceptadas ou falsificadas por *malwares*, outra falha presente nas API's, é a de que os dados dos aplicativos que são removidos do cartão de memória continuarem armazenados nele. Também uma problemática existente nas API's é a utilização de excesso de permissões autorizadas para aplicativos, sendo que cerca de 22% dos aplicativos necessitam acesso do IMEI (*International Equipament Identity*) sendo que alguns aplicativos enviam dados do IMEI de forma não criptografada. Segundo Javvin (2007), IMEI é um número de identificação único e global para cada dispositivo móvel, ele é usado pela rede GSM para identificar se os dispositivos são válidos, podendo bloqueá-lo em caso de perda ou roubo. Estes problemas relacionados a segurança das API's são quase que totalmente responsabilidade de desenvolvedores que muitas vezes não programam de forma segura suas aplicações e também da mantenedora da Plataforma *Android*.

O *Bluetooth* também representa um risco ao sistema. Isso se deve ao fato de o serviço de *bluetooth* presente no caminho "com/android/phone/BluetoothHeadset

[12] http://seumicroseguro.com/?s=SMS+e+MMS
[13] http://www.mcafee.com/br/resources/reports/rp-securing-mobile-devices.pdf

Service.java", nas versões anteriores á 2.3.6 da Plataforma *Android*, permite ao atacante a possibilidade de obter dados por meio do ataque realizado por uma transferência de comandos AT, onde estes são dados para a configuração do dispositivo via *bluetooth* para a agenda do telefone, segundo o *Android*, esta vulnerabilidade foi resolvida a partir da versão 2.3.6.

Outro problema de segurança é a de escalada de privilégios no *kernel Linux*. Segundo o site phandroid (2011)[14], uma vez que qualquer aplicativo sem privilégios maiores pode ganhar privilégios e tomar controle do dispositivo obtendo privilégio do superusuário *root*, ou seja, um usuário pode instalar um aplicativo e este pode tomar conta do dispositivo, podendo gerencia-lo totalmente, já que ele obteve permissões de uso do superusuário *root*.

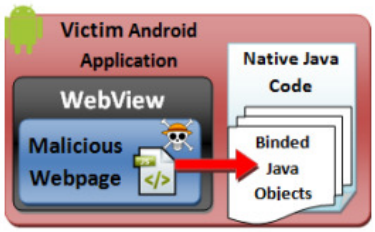
O browser utilizado pela Plataforma *Android* não restringe corretamente *cookies* em sessões HTTPS, podendo ocasionar ataques *man-in-the-middle*[15], onde os *cookies* são apagados ou substituídos. Este ataque ocorre enviando um cabeçalho *Set-Cookie*, no qual são definido os cookies que serão enviados no cabeçalho HTTP[16], relacionada a falta de HSTS (*HTTP Strict Transport Security*), que é um mecanismo que força ao browser sempre utilizar criptografia nos acessos HTTPS.

A *Web View* é uma classe utilizada para aplicações *Android* que necessitem utilizar um *browser* em seu conteúdo, ou utilizar algum conteúdo online no aplicativo. A maioria dos aplicativos que estão entre os mais utilizados por usuários da plataforma *Android* utilizam esta classe, entre os tipos de ataques que ocorrem utilizando a classe *Web View* estão:

- Ataques de *WebPages* mal-intencionados: Neste tipo de ataque as aplicações presentes no dispositivo *Android*, são alvos de *webPages* maliciosas, onde o atacante necessita que o usuário as carregue no aplicativo, podendo assim atacar a *webView* da aplicação, que contem vários recursos, inclusive com os dados do aplicativo e até do usuário, sendo que para carregar esta *webPage*, não é tão difícil, pois ela pode estar presente em um email, rede social, propagandas e etc. A seguir é possível visualizar como ocorre este tipo ataque através da figura 7:

[14] http://phandroid.com/2011/09/20/2-android-bugs-allow-malicious-apps-to-install-on-your-device-without-your-permission/
[15] http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html
[16] http://php.net/manual/pt_BR/function.setcookie.php

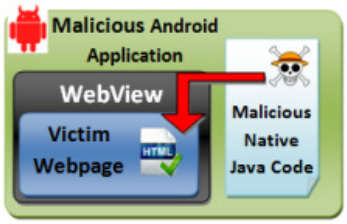
Figura 7 - Ataque á webView por páginas web



Fonte: http://www.cis.syr.edu/~wedu/Research/paper/webview_acsac2011.pdf

- Ataques de Aplicativos mal-intencionados: Neste tipo de ataque as aplicações web são os alvos, sendo que para ocorrer este ataque é necessário que usuários utilizem a aplicação do atacante para a aplicação web alvo. A *Web View* fornece muitas formas de interagir com os aplicativos do *Android*, promovendo uma interação melhor com as páginas *web*, isto possibilita aos atacantes utilizarem *sniffers*, podendo interceptar as APIs e cometer ataques fora da *Web View*. Na figura 8 é possível visualizar este tipo de ataque:

Figura 8 - Ataque á Web View de aplicativos mal-intencionados



Fonte: http://www.cis.syr.edu/~wedu/Research/paper/webview_acsac2011.pdf

6 CRITÉRIOS PARA OS TESTES DE FERRAMENTAS DE SEGURANÇA

Para que seja possível realizar os testes com as ferramentas de segurança, é necessário inicialmente impor alguns critérios, para que estes testes sejam eficientes e corretos. A seguir serão descritos alguns dos critérios escolhidos para a realização dos testes, sendo que os critérios foram divididos em três categorias, a de antivírus, *firewall* e VPN. Estes critérios foram baseados em pesquisas de testes de softwares de segurança, divulgadas em revistas conceituadas na área de T.I. . Também foi levado em consideração trabalhos científico da área de segurança na realização de testes semelhantes.

6.1 CRITÉRIOS UTILIZADOS PARA OS TESTES COM ANTIVÍRUS

1. Eficiência: No critério de eficiência, será avaliada a capacidade da ferramenta de segurança, em detectar ataques no dispositivo móvel. Para os ataques, serão utilizados, programas e técnicas para a criação das vulnerabilidades.
2. Desempenho: Para o critério de desempenho, será avaliado a performance quanto ao custo computacional que a ferramenta necessita utilizar, uma vez que o dispositivo móvel possui recursos de hardware inferior ao de um PC, sendo que qualquer consumo excessivo de hardware, pode comprometer o funcionamento normal do dispositivo. Para a análise do desempenho, será averiguado, o consumo de memória, armazenamento, CPU e bateria.
3. Pontos Positivos: Serão avaliados os recursos oferecidos pela ferramenta de segurança, levando em consideração os itens básicos de uma ferramenta de segurança como a detecção em tempo real, sem necessitar realizar varredura para detectar a invasão, além de outros serviços disponibilizados pela ferramenta.
4. Deficiências: Também serão levadas em consideração as deficiências da ferramenta, como a ausência de recursos essenciais em uma ferramenta de segurança.

5. Qualidade de diagnóstico: será levado em consideração a forma como a ferramenta apresenta o diagnostico da varredura.
6. Tempo de resposta da ferramenta: Neste item, será avaliado o tempo em que a ferramenta irá utilizar para realizar as tarefas que serão demandadas.

6.2 CRITÉRIOS UTILIZADOS PARA OS TESTES COM *FIREWALL*

1. Eficiência: A capacidade da ferramenta de firewall conseguir bloquear o ataque realizado. Para os ataques, serão utilizados, programas e técnicas para a criação das vulnerabilidades.
2. Desempenho: Para o critério de desempenho, será avaliado a performance quanto ao custo computacional que a ferramenta necessita utilizar, uma vez que o dispositivo móvel possui recursos de hardware inferior ao de um PC, sendo que qualquer consumo excessivo de hardware, pode comprometer o funcionamento normal do dispositivo. Para a análise do desempenho, será averiguado, o consumo de memória, armazenamento, CPU e bateria.
3. Pontos Positivos: Serão avaliados os recursos oferecidos pela ferramenta de segurança, levando em consideração os itens básicos de uma ferramenta de segurança, como firewall bloquear acesso de ataques mal-intencionados, bloqueio de acesso para um aplicativo, além de oferecer opção de criar as próprias regras do usuário.
4. Deficiências: Também serão levadas em consideração as deficiências da ferramenta, como a ausência de recursos essenciais em uma ferramenta de segurança, como o bloqueio de aplicações com permissão de utilizar acesso ao *root*, bloqueio de aplicações para acesso ao *kernel*, bloqueio para qualquer aplicação e suporte para diferentes línguas.

6.3 CRITÉRIOS UTILIZADOS PARA OS TESTES COM VPN

1. Desempenho: Para o critério de desempenho, será avaliado a performance quanto ao custo computacional que a ferramenta necessita utilizar, uma vez que o dispositivo móvel possui recursos de hardware inferior ao de um PC, sendo que qualquer consumo excessivo de hardware, pode comprometer o funcionamento normal do dispositivo. Para a análise do desempenho, será averiguado, o consumo de memória, armazenamento, CPU e bateria.
2. Pontos Positivos: Serão avaliados os recursos oferecidos pela ferramenta de segurança, levando em consideração os itens básicos de uma ferramenta de segurança, como VPN ocultar endereço IP da máquina e codificação de dados.
3. Deficiências: Também serão levadas em consideração as deficiências da ferramenta, não possuir ocultação de endereço IP, não codificar dados e se a ferramenta não conseguiu estabelecer conexão.

Para a averiguação do desempenho das ferramentas de segurança testadas, será utilizada a ferramenta *Android Assistant*, disponibilizada na loja *Google Play*, esta ferramenta foi criada para melhorar o funcionamento do dispositivo móvel, possibilitando o monitoramento da CPU, memória, *SDCard*, bateria, dentre outros serviços. Toda a análise realizada em cima das ferramentas de segurança terá como base os critérios listados acima, para que no fim dos testes, possa haver um comparativo entre elas.

7 TESTE DAS FERRAMENTAS DE SEGURANÇA *ANDROID*

A seguir serão descritos os passos realizados para a execução do teste de invasão, como: a metodologia adotada, a execução do teste e seus resultados. Primeiramente será explicada a metodologia utilizada para a realização dos testes de invasão, em seguida a seleção de ferramentas para a execução dos testes, em seguida como os testes de invasão foram executados.

7.1 METODOLOGIA ADOTADA NO TESTE DE SEGURANÇA

Para que os testes a serem realizados possam resultar em conclusões satisfatórias, é necessário utilizar metodologias adotadas em testes de segurança. Alguns dos itens da metodologia de testes, estão presentes no manual da NIST (*National Institute of Standart and Technology*)[17] , o tipo de teste mais direcionado para o contexto do trabalho, é o teste de invasão. Conforme Uto, "[...] é um método utilizado para verificar a segurança de um ambiente, plataforma ou sistema, por meio de simulação de ataques reais explorando as vulnerabilidades encontradas." (2011, p.39).

Segundo a NIST, o teste de invasão deve ser dividido em quatro fases:

1º Planejamento: Nesta fase, o foco é o planejamento do teste de invasão, definindo objetivos e ferramentas para a execução do teste.

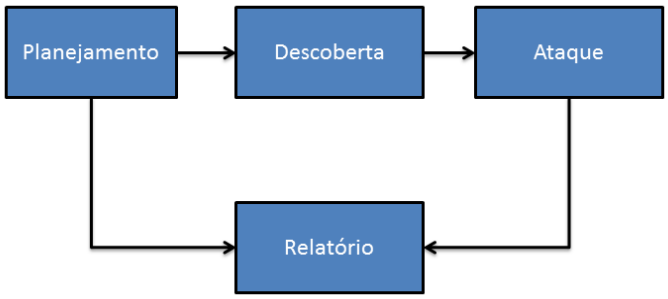
2º Descoberta: Fase realizada em duas etapas, sendo que na primeira, ocorre a coleta de informações sobre o sistema e das ferramentas de segurança, identificando as vulnerabilidades existentes. Na segunda etapa, as vulnerabilidades identificadas são avaliadas.

3º Ataque: Principal fase do teste de invasão, sendo que as vulnerabilidades encontradas na fase Descoberta são exploradas utilizando programas e ferramentas direcionadas a ataques em sistemas.

4º Relatório: Fase em que é executada em paralelo com as três fases anteriores, sendo que nesta fase ocorre a formação de relatórios, com os resultados correspondentes ao teste, sendo que ao fim deve ser formado um relatório completo sobre toda a execução do teste, descrevendo o nível de segurança do sistema.

[17] NIST(*National Institute of Standart and Technology*) – O instituto criou o Manual chamado "Guia Técnico para Avaliações e Testes de Segurança da Informação", onde é definido metodologias de testes de segurança em ambientes corporativos, bem como testes de invasão, com simulações de ataques reais. Disponível em:< http://pt.scribd.com/doc/65629231/Plugin-Artigo-Pentest-Cristiano >

Figura 9 - Fases do teste de invasão



Fonte: Adaptado de NIST, 2008, p.53

As quatro fases mostradas na figura 9 serão levadas como referência para a execução dos testes, tendo em vista que para a comparação entre as ferramentas, se levará em consideração os critérios estipulados listados no capítulo 6.

7.2 SELEÇÃO DAS FERRAMENTAS DE SEGURANÇA

Para a fase de planejamento, foram selecionadas as ferramentas mais utilizadas segundo o site de aplicativos da Plataforma *Android*[18], e também as ferramentas mais bem conceituadas pelo teste realizado pela empresa AVTEST[19]. Os testes realizados tiveram como objetivo identificar se as ferramentas são capazes de detectar e proteger o dispositivo dos ataques, para isso, as ferramentas foram divididas em três categorias: antivírus, *firewall* e VPN. Uma vez que os ataques realizados possuem diferentes focos, sendo que para uns ataques as ferramentas antivírus são mais adequadas e para outros o firewall e outros VPN também levando em consideração quais os serviços que cada ferramenta disponibiliza para o usuário.

7.2.1 Ferramentas de Antivírus

Os antivírus mais utilizados pelos usuários e mais bem colocados nos testes realizados pelo instituto AVTEST são:

[18]https://play.google.com

[19] AVTEST: Instituto alemão que possui como foco detectar malwares, utilizando análise e técnicas, tendo como clientes empresas fabricantes de ferramentas de segurança. Disponível em: http://www.av-test.org

- AVG Antivírus - *Free*: Antivírus gratuito, possuindo 555.194 downloads em junho de 2013. Possui serviços que prometem a proteção contra vírus*, malware, spyware* e exploração *online*. Também segurança contra furto do dispositivo, navegação segura, monitoramento do desempenho do dispositivo e apagar completamente o histórico do dispositivo, como mensagens fotos e etc. Segundo o site de aplicativos da Plataforma Android[20], entre seus recursos oferecidos podem ser destacados: Atualização, verificação automática por dia, semana ou nunca, verificação de arquivos por categoria: *SDCard, Root, Pictures, Music* e *Video*. Navegação segura em *websites* suspeitos, verificação de mensagens de texto. Aumentar o desempenho do celular: *Task Killer,* para fechar aplicativos não-utilizados, Consumo de Bateria, para otimizar a utilização da bateria, utilização de dados móveis e gerenciamento do armazenamento do dispositivo. Recurso Antifurto, com a capacidade de localizar o dispositivo via *google maps*, possuindo comandos para ativar alarme remotamente. Serviço de Privacidade com definição de senha para o dispositivo, *backup* de aplicativos instalados no *SDCard*, limpar dados do dispositivo, limpar dados por categoria e formatar cartão SD.

- *Lookout Security* & Antivírus: Antivírus criado pela empresa Lookout, possui versão *free* e versão paga, possuindo 434,733 de downloads em junho de 2013, promete proteção contra *malware, spyware* e *trojans*, verificação de aplicativos baixados e monitoramento do sistema de arquivo contra *malware*. Segundo o site de aplicativos da Plataforma Android[21], entre seus recursos oferecidos podem ser destacados: Verificação automática, *backup* de contatos, fotos e histórico de chamada, dispositivo perdido por GPS, navegação segura, analisador de apps, definindo quais as permissões que os apps utilizam do dispositivo.

- *Dr Web* Antivírus *Light*: Antivírus desenvolvido pela empresa *Doctor Web Ltd*, sendo considerada um dos antivírus com melhor performance pelo teste da empresa AVTESTE, possui versão gratuita e paga, com 139.794 downloads em junho de 2013, promete proteção contra vírus e spam, também proteção cartão SD contra contaminação de *autorun* e *exploit.cllnk*. Segundo o site de aplicativos da Plataforma *Android*[22], entre seus recursos oferecidos**:** Varredura rápida ou completa do sistemas de arquivos, detectar novos

[20] https://play.google.com/
[21] https://play.google.com/
[22] https://play.google.com/

malwares, proteção do cartão SD, quarentena para ameaças, impacto mínimo no desempenho do S.O., interface de fácil uso.

- *Avast!Mobile Security*: Criado pela empresa *Avast Software*, com 318.104 *downloads* em junho de 2013, promete proteção antivírus e antirroubo, foi o aplicativo antivírus, mais bem colocado pelo teste realizado pela empresa AVTESTE. Segundo o site de aplicativos da Plataforma *Android*[23], entre recursos oferecidos**,** estão proteção em tempo real do dispositivo, possibilidade de escaneamento automático, definindo por dia da semana e horário, possibilita escaneamento de programas e cartão, monitora as permissões de acesso dos apps, gerenciador de programas obtendo, informações do programa como uso de memória e forçar interrupção do app, controle de módulos de aplicativos, *internet* e mensagens, filtrar SMS e chamadas, *firewall*, medidor de tráfego por 3g, *wifi*, *roaming* e anti-furto.

- Norton *Mobile Security Lite*: Possui cerca de 117.705 de *downloads* em junho de 2013, este antivírus promete proteção contra *malware* e vírus, análise automática de novas transferências e atualizações de aplicações, bloqueia mensagens SMS indesejáveis, impede web sites fraudulentos de acesso á informações pessoais do usuário, dentre outros serviços. Segundo o site de aplicativos da Plataforma *Android*[24],entre os recursos oferecidos**,** estão enviar mensagens SMS para desbloqueio do dispositivo em caso de perda ou extravio, análise e remoção de aplicativos e atualizações que tenham a intenção de prejudicar o dispositivo, análise do *SDCard*.

- ESET *Mobile Security*: Promete proteção contra ataques via *wi-fi* públicas, proteção *antimalware*, rastreando arquivos de aplicativos baixados e acessados, bloqueio de SMS de números desconhecidos. Possui 8.234 downloads em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[25], entre os recursos oferecidos, estão recursos antifurto, com bloqueio remoto, limpeza remota e localização do dispositivo. Proteção *antimalware*, proteção *antispam* SMS, auditoria de segurança com gerenciador de tarefas e suporte para diferentes idiomas.

[23] https://play.google.com/
[24]https://play.google.com/
[25]https://play.google.com/

7.2.2 Ferramentas de *Firewall*:

As ferramentas de *firewall* foram escolhidas conforme a sua utilização pelos usuários da Plataforma *Android*, levando em consideração os mais baixados, entre eles estão:

- *DroidWall – Android Firewall*: *firewall* iptables do Linux, promote restringir quais aplicativos podem acessar a rede de dados, também é possível criar regras com este *firewall*, com 9.304 downloads em junho de 2013, é uma ferramenta gratuita. Segundo o site de aplicativos da Plataforma *Android*[26], entre os recursos oferecidos, estão restringir aplicativos para utilização da rede, tanto wi-fi quanto 2G/3G e criação de regras customizadas.

- *Bluetooth Firewall Trial*: Criado pela empresa *FruitMobile*, é uma ferramenta paga e promete proteção, impossibilitando a porta aberta do *bluetooth* contra ataques de *hackers*. Possui 148 downloads em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[27], entre os recursos oferecidos, está proteção para o *bluetooth* do dispositivo.

- *Root Firewall*: Criado pela empresa *Root Unistaller*, promete bloqueio de propagandas e salvar a vida da bateria, possui 753 downloads em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[28], entre os recursos oferecidos, estão bloqueio de acesso á internet por qualquer aplicação, separação de 3G e dados *wifi, widget* para habilitar com um clique, bloqueio de aplicativo com um toque.

- Avast *Mobile Security*: Criado pela empresa Avast *Software*, com 318.104 downloads em junho de 2013, além deste mesmo aplicativo possuir antivírus, possui *firewall* integrado. Segundo o site de aplicativos da Plataforma *Android*[29], entre os recursos oferecidos, estão o bloqueio de aplicações para utilização da rede wifi, bloqueio de aplicações com permissão de utilizar *root* de acessar a rede, bloqueio de aplicações com acesso ao *kernel* de utilizar a rede e criação de regras especificas de firewall.

[26] https://play.google.com/
[27] https://play.google.com/
[28] https://play.google.com/
[29] https://play.google.com/

7.2.3 Ferramentas de VPN

As ferramentas de VPN foram escolhidas conforme a sua utilização pelos usuários da Plataforma *Android*, levando em consideração os mais baixados, entre eles estão:

- Viatun 4 VPN: Serviço VPN, criado pela empresa Viatun *Software*, promete a redução de tráfego em 80%, tornar a internet móvel segura e anônima, além de proteção antivírus, com 1.865 *downloads* em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[30], recursos oferecidos, estão redução de gastos com tráfego de dados em até 80%, internet segura e anônima, ocultação do IP para navegação anônima, proteção do dispositivo em conexões wifi públicas.

- VPN *One Click*: Criado pela empresa Bravotelco LLC, promete segurança na *web*, com 4.206 downloads em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[31], dentre os recursos oferecidos estão modificar o endereço IP do dispositivo tornando diferente do real e conexão com a internet criptografada.

- *Hideman* VPN: Criado pela empresa *Hideman* Ltd, possui acesso gratuito 5 horas por semana, possui 10.391 downloads em junho de 2013. Segundo o site de aplicativos da Plataforma *Android*[32], entre os recursos oferecidos, estão ocultar o endereço IP do dispositivo, codificação de dados.

Todas as ferramentas selecionadas possuem em seus serviços formas de proteger os dispositivos quanto aos ataques realizados, será avaliado se as ferramentas conseguem interceptar os ataques e se cumprem com o prometido por elas mesmas aos seus usuários.

7.3 EXECUÇÃO DO TESTE DE INVASÃO

Para realizar os teste de invasão, foi utilizado o BackTrack 5, que é uma distribuição Linux inteiramente focada em testes de invasão, possuindo cerca de 300 ferramentas para a execução da mesma. Segundo Giavaroto e Santos (2013, p.5), ele é

[30] https://play.google.com/
[31] https://play.google.com/
[32] https://play.google.com/

> Baseado no WHAX, Whoppix e Auditor, Backtrack é uma ferramenta voltada para testes de penetração muito utilizada por auditores, analistas de segurança de redes e sistemas, hackers éticos etc. Sua primeira versão é de 26 de maio de 2006.[...] Atualmente, possui mais de 300 ferramentas voltadas para testes de penetração, existem ainda algumas certificações que utilizam o Backtrack como ferramenta principal, OSCP Offensive Security Certified Expert e OSWP Offensive Security Wirelless Professional, certificações oferecidas pela Offensive Security que mantém o Backtrack.

Dentre as ferramentas presentes no BackTrack, serão utilizadas no teste de invasão aquelas específicas para as ferramentas destacadas anteriormente.

Para a invasão por *Bluetooth* será utilizado o *Bluediving*, que possui uma suíte de testes de invasão para *bluethooth*, sendo que o utilizado para realizar o teste será o *BlueBug* AT *Shell*, segundo a trifinite[33], *bluebugg* é um tipo vulnerabilidade presente em alguns dispositivos pelo *bluetooth*, sendo que uma vez que exista esta vulnerabilidade, é possível ocorrer ataques ao dispositivo, como os contatos do telefone, chamadas realizadas, leitura de SMS recebidos, bem como a leitura de SMS enviados, além de alguns outros ataques.

Para o ataque via rede wi-fi, será utilizado o Dsniff, que possui uma coleção de ferramentas para a realização de testes em redes, sendo que a ferramenta selecionada para a realização dos teste, foi a *Arpspoof*, que ”é uma ferramenta de *sniffing* de uso geral[...] Ela entente vários serviços diferentes que transmitem informações de senha em texto claro, além de outras se você lhe der a chave apropriada”. (CHESWICK; BELOVIN; RUBIN, 2003, p.138). Também foi utilizada a ferramenta diftnet, ela é capaz de escutar tráfego de rede e escolher imagens de *streams* TCP por ela encontrada.

Para o ataque via browser, a ferramenta utilizada foi o Metasploit, ela tem como foco ganhar o acesso ao sistema atacado através de *exploit* e *payload*, vistos anteriormente.

Para realizar os testes por aplicativos *Android*, foram utilizadas a técnica de engenharia reversa, que tem como foco a utilização de aplicações prontas, porém estas são convertidas em código, sendo que atacantes as utilizam para inserir código com intenção de ganhar acesso ao aparelho, para esta técnica não foi necessário o uso de nenhuma ferramenta presente no backtrack5, mas apenas técnicas de programação.

Para os testes com envio de SMS, os aplicativos de segurança prometem o bloqueio de envio de SMS por telefones não seguros, para isso foi utilizado outro dispositivo para realizar o envio de SMS.

---

[33] Trifinite: Organização sem fins lucrativos, formado por especialistas em TI, que realizam pesquisas com comunicações sem fio. Disponível em: <http://trifinite.org/trifinite_org.html>.

Os dispositivos utilizados para a realização dos testes possuem as seguintes configurações, obtidas através do dispositivo em suas configurações:

Tabela 1 - Informações dos dispositivos móveis

Dispositivo Atacado:	
Número do Modelo	GT-S5360B
Fabricante	Samsung
Modelo	Galaxy Y
Versão Android	2.3.5
Versão Kernel	2.6.35.7
Número de compilação	GINGERBREAD.UTKJ2
Dispositivo envia mensagens SMS:	
Número do Modelo	ViewPad7
Fabricante	ViewSonic
Modelo	ViewPad7
Versão Android	2.2.1
Versão Kernel	2.6.32.7
Número de compilação	1008_3_240

Fonte: da Autora

Além da utilização de dois dispositivos para realizar os testes, também foi utilizada uma máquina virtual instalada o S.O. Backtrack 5, possuindo as seguintes configurações:

Tabela 2 - Configuração da máquina virtual

Configuração máquina virtual	
Memória RAM	1 GB
Armazenamento	16GB
Rede	Placa em modo Bridge

Fonte: da Autora

Os testes foram executados de forma homogênea para todos os aplicativos de segurança testados, sem que o dispositivo sofresse qualquer modificação na realização dos testes, como a instalação de novos aplicativos ou a inserção de novos arquivos no dispositivo, sendo que estes podem intervir no tempo de varredura dos aplicativos de segurança.

7.3.1 Ataque via Bluetooth

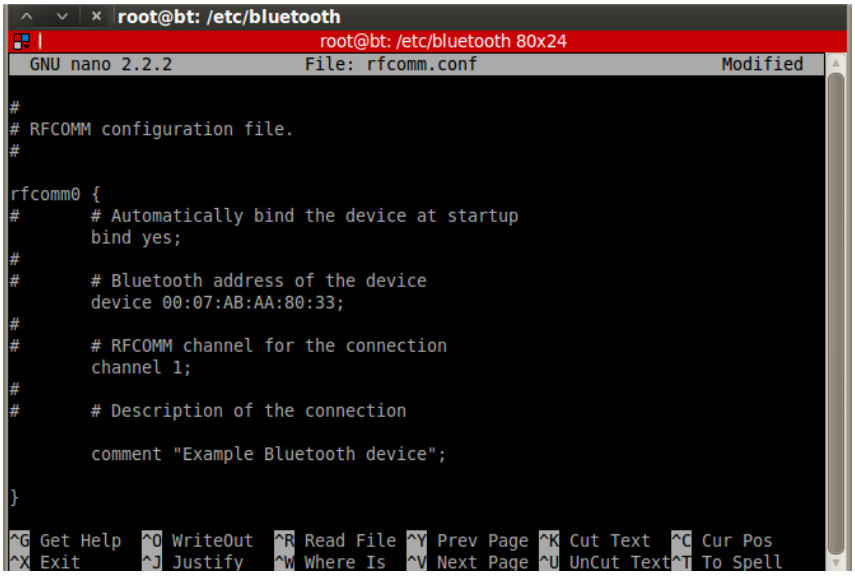
O *bluetooth* está presente nos dispositivos móveis e é um dos alvos favoritos de atacantes, com ele é possível realizar conexões entre dispositivos que possuam *bluetooth* para o compartilhamento de arquivos. É especificada como rede pessoal sem fio ou WPAN (*Wireless Personal Area Network*).

Para a realização do ataque por *bluetooth*, foi necessário a configuração do RFCOMM (Protocolo de Comunicação de Radiofrequência). Segundo Stallings (2005, p.239)

> RFCOMM é o protocolo de substituição de cabo incluído na especificação Bluetooth. O RFCOMM apresenta uma porta serial virtual que é projetada para tornar a substituição das tecnologias de cabo o mais transparente possível. As portas seriais são um dos tipos mais comuns de interfaces de comunicação usadas em dispositivos de computação e de comunicação.

Com o comando **# mknod -m 666 /dev/rfcomm0 c 216 0** inserido no terminal é possível criar os dispositivos do RFCOMM, uma vez que estes não são automaticamente montados pelo sistema, a partir da criação deste dispositivo poderá ocorrer a conexão com o dispositivo móvel e o computador. Em seguida é necessária a configuração do arquivo rfcomm.conf, que fica localizado no diretório /etc/bluetooth, este arquivo possui as configurações para criar um *link* entre o dispositivo móvel e o computador, as seguintes modificações foram realizadas:

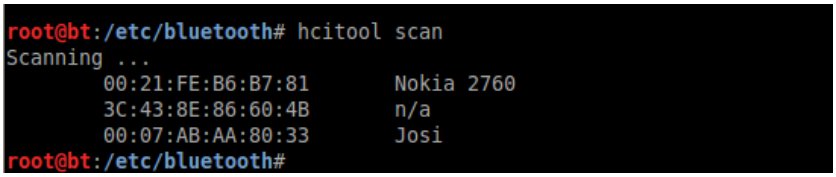
Figura 10 - Configuração do arquivo rfcomm.conf



Fonte: da Autora

Podemos verificar as modificações realizadas no arquivo pela figura 10, onde foram descomentadas todas as linhas de configuração, foi modificada a linha *bind* de "*no*" para "*yes*", a linha *device*, onde foi inserido número mac do *bluetooth* do telefone, que pode ser obtido através do comando **#hcitool scan**, na figura 11, é possível visualizar o resultado do comando:

Figura 11- Comando scan



Fonte: da Autora

Foi scaneado o numero *mac* de dispositivos *bluetooth*, e em seguida o nome do dispositivo, também foi modificado o numero da linha *Chanel* do arquivo rfcomm.conf, inserindo o número 1. Segundo Morimoto (2008), é o canal de comunicação mais usado por alguns dispositivos móveis, existindo exceções em alguns dispositivos. Em seguida foi realizada a conexão do dispositivo *bluetooth* com o computador através do comando **#rfcomm bind /dev/rfcomm 00:07:AB:AA:80:33 1**, onde /dev/rfcomm simboliza a localização do *device* criado em seguida o número *mac* do dispositivo *bluetooth* e o canal de conexão. Na figura 12, é demonstrado como averiguar se ocorreu a conexão com o dispositivo, através do comando **#rfcomm.**

Figura 12 - Comando rfcomm

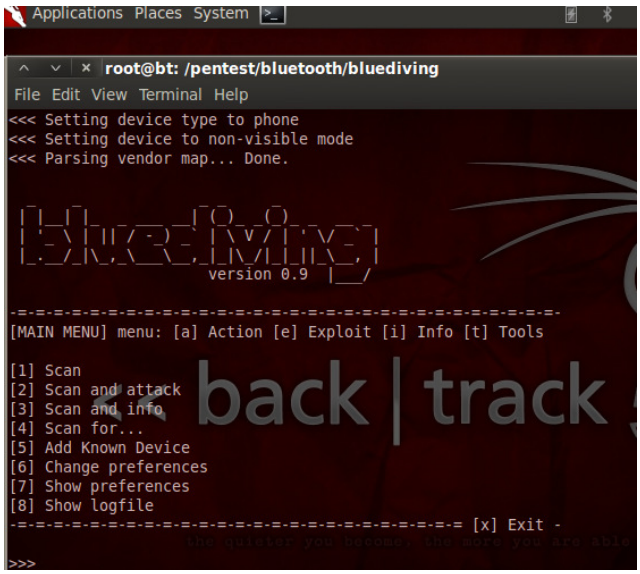
Fonte: da Autora

Após a configuração da conexão entre o computador e o dispositivo, é necessário realizar o pareamento dos dispositivos, segundo Morimoto(2008), o passkey, que é o código de segurança para realizar o pareamento entre dispositivos através de *bluetooth*, tem como padrão o valor "1234", normalmente os usuários de dispositivos, não modificam esta chave, dando aos atacantes a chance de parear com o dispositivo atacado.

Para realizar o ataque foi utilizado a ferramenta *Bluediving*, presente no arsenal de ferramentas do BackTrack 5, esta ferramenta contêm um suíte de testes de invasão por *bluetooth*, a pasta com a ferramenta encontra-se no diretório "/pentest/bluetooth/bluediving",

onde possui um *script* desenvolvido em linguagem *Pearl* denominado “bluedivingNG.pl”, para executá-lo foi inserido o comando #./bluedivingNG.pl, em seguida uma tela é formada com opções a serem inseridas, conforme a figura 13:

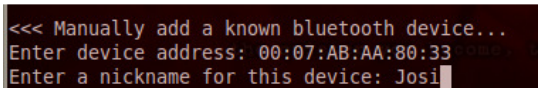
Figura 13 - Tela ferramenta Bluediving



Fonte: da Autora

Em seguida foi inserido a opção 5, conforme demonstra a figura 13, onde é possível adicionar manualmente um dispositivo para ser atacado, como o dispositivo já havia sido identificado anteriormente não foi necessário um novo escaneamento, a figura 14 demonstra as informações inseridas, no caso o número *MAC* do dispositivo atacado e o *nickname* ou apelido do dispositivo:

Figura 14 - Dados do dispositivo atacado

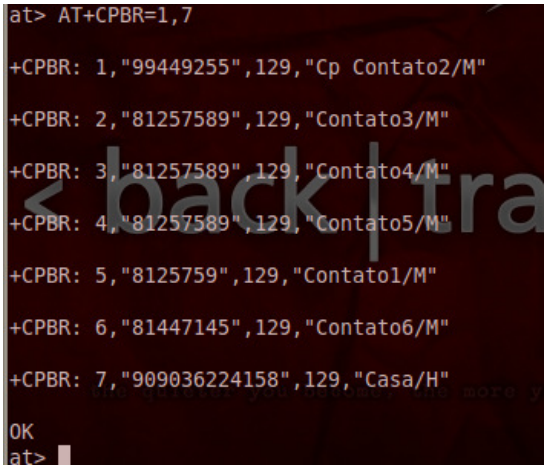


Fonte: da Autora

Em seguida foi inserido a opção “e” para realizar o ataque *Exploit*, a ferramenta usada foi o *Blue Bug AT shell*, com esta ferramenta é possível cometer ataques ao dispositivo sem o usuário perceber, pois aproveita brechas de segurança permitindo o atacante a ter acesso à lista de contatos, ler mensagens, além de outros ataques. Esta ferramenta utiliza comandos AT, que é uma forma de obter uma comunicação entre o *bluetooth* e o computador, enviando comandos para configurar ou obter dados, primeiramente foi inserido o comando “at > AT+CPBR=?”, para verificar quantos contatos o dispositivo possui em seguida o comando

"AT+CPBR=1,7", onde "1,7" define que deverá ser feita a leitura dos contatos 1 até o 7. A figura 15 demonstra o resultado alcançado, pode-se visualizar que os sete contatos foram escaneados, mostrando o número de contato e nome, sendo que o dispositivo atacado não obteve conhecimento do ataque:

Figura 15 - Resultado do ataque realizado por comando AT



Fonte: da Autora

7.3.2 Ataque Via Rede *Wi-fi*

A utilização da internet móvel por dispositivos móveis é considerada quase que indispensável ainda mais com a oferta de redes abertas em shoppings, aeroportos e rodoviárias sendo quase impossível permanecer "desconectado". Os atacantes também aproveitam para usufruir de vulnerabilidades presentes neste ambiente, a seguir será demonstrado como é possível realizar este tipo de ataque para obter dados de usuário de dispositivos móveis.

Primeiramente foi necessário habilitar o encaminhamento de pacotes, habilitando o ip_forward, sendo necessário inserir o comando **#echo 1 > /proc/sys/net/ipv4/ip_forward** no terminal como superusuário. Segundo Maquione (2009)[34], "Este comando escreve o número 1 dentro do arquivo ip_forward, ativando o roteamento de pacotes de uma interface para a outra e vice-versa.".

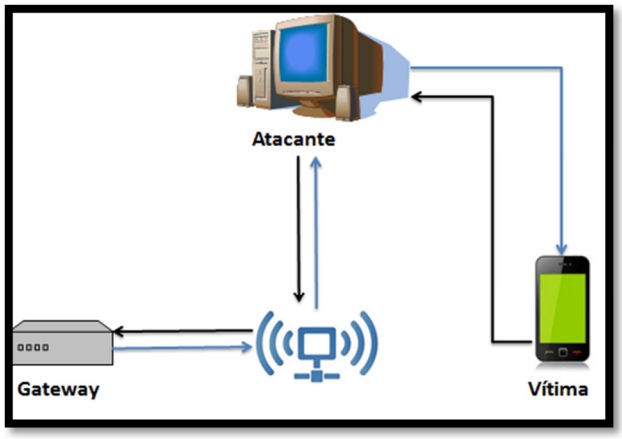
Após foi necessário utilizar uma ferramenta presente no dsniff, o *arpspoof*, com ele é possível redirecionar os pacotes, considerado como um ARP *poison*, para a realização deste

[34] http://www.vivaolinux.com.br/dica/Roteamento-de-pacotes-e-NAT-no-Linux

ataque. Foi necessário utilizar dois comandos e em dois terminais, **#arpspoof –i eth0 –t 192.168.1.101 192.168.1.1** e **#arpsoof –i eth0 –t 192.168.1.1 192.168.1.101.**

Com estes dois comandos os pacotes enviados para a vítima são enviados para a máquina atacante, e os pacotes enviados para o *gateway*[35], ao invés de serem enviados á vitima são enviados á máquina atacante, sendo que 192.168.1.101 é o numero ip do dispositivo móvel e 192.168.1.1 é o número do *gateway*. Porém estes pacotes são reenviados para o *gateway*, assim a rede não ficará desabilitada e o ataque não será percebido. Na figura 16 é possível visualizar como este ataque ocorre:

Figura 16 - Ataque arpspoof



Fonte: da Autora

Assim todos os pacotes passam pelo atacante, podendo ele capturar e obter dados sigilosos da vitima. Para finalizar o ataque, foi utilizado a ferramenta driftnet, esta ferramenta tem a função de encontrar imagens em sessões TCP, com ele é possível averiguar quais imagens a vítima está acessando, o comando para executar esta ferramenta é **#driftnet –i eth0**.

Assim é aberta uma janela com as imagens acessadas pela vitima, como pode ser visualizado na figura 17, onde a vitima estava acessando uma página com imagens do Backtrack.

[35] Gateway: Pode ser dito como uma ponte de ligação, podendo ser interligar redes, podendo haver comunicação entre diferentes arquiteturas e ambientes. Disponível em:< http://pt.wikipedia.org /wiki/Gateway>.

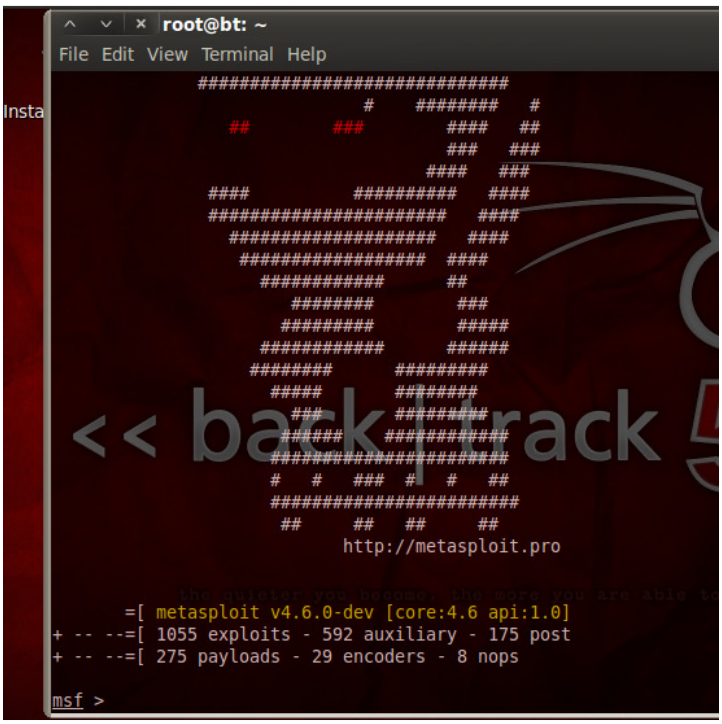
Figura 17 - Imagem driftnet

Fonte: da Autora

7.3.3 Ataque via Browser

Para a realização deste ataque, foi necessário primeiramente abrir a ferramenta *metasploit* com terminal no S.O. Backtrack através do comando **#msfconsole,** a figura 18 apresenta a tela inicial da ferramenta**:**

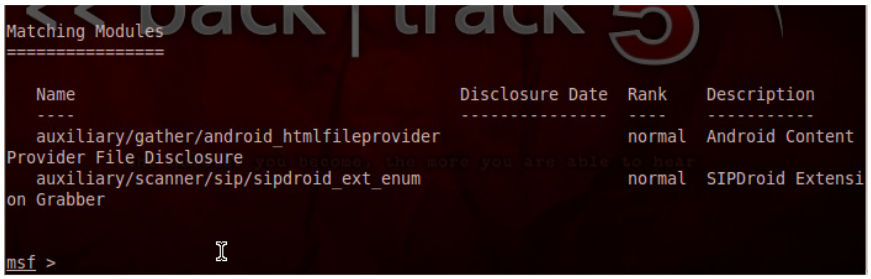
Figura 18 - Imagem da tela inicial da ferramenta metasploit



Fonte: da Autora

Após foi inserido o comando **msf > search android,** com a intenção de encontrar o ataque para a plataforma *Android*, os seguintes módulos foram encontrados conforme mostrado na figura 19:

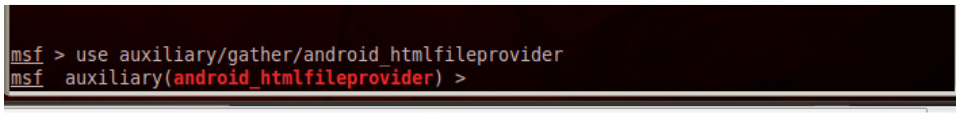
Figura 19 - Módulos encontrados para Plataforma Android



Fonte: da Autora

Para o teste o módulo utilizado foi o *auxiliary/gather/android_htmlfileprovider*, com este módulo é possível enviar ao dispositivo um arquivo com extensão html. Em seguida foi inserido o comando **msf > use auxiliary/gather/android_htmlfileprovider**, conforme a figura 20. Este comando define para a ferramenta, que será utilizado o módulo *auxiliary/gather/android_htmlfileprovider*.

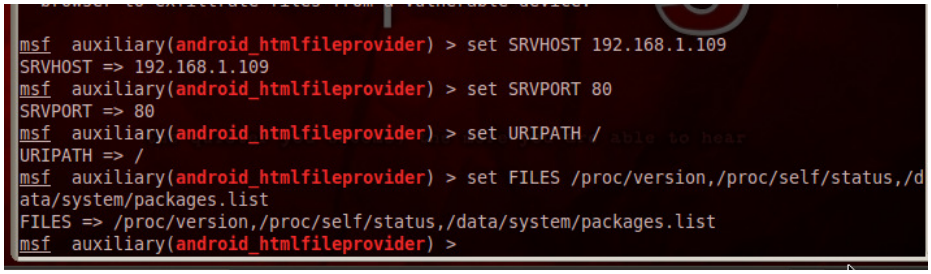
Figura 20 - Módulo usado no ataque



Fonte: da Autora

Após foi necessária a inserção de alguns comandos com os dados da máquina do atacante, conforme definido na figura 21:

Figura 21 - Comandos da máquina do atacante

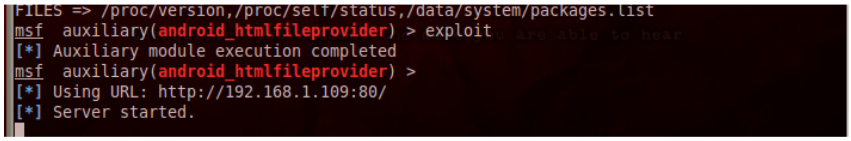


Fonte: da Autora

O primeiro comando, **set SRVHOST 192.168.1.109,** identifica o número IP da máquina do atacante, o segundo comando **set SRVPORT 80**, a porta em que o ataque será realizado, como o alvo irá utilizar o browser para acessar o atacante, a porta 80 é a porta padrão para o protocolo HTTP, o terceiro comando se**t URIPATH /,** define que a URI do

ataque ocorrerá sem nenhuma necessidade de inserir qualquer outra palavra após a inserção do número IP da máquina do atacante no *browser* do alvo. E no último comando **set FILES /proc/version,/proc/self/status,/data/system/packages.list,** identifica quais os arquivos que serão acessados pelo atacante quando ocorrer o acesso do alvo no browser. Este caminho indica acesso em todas as pastas presentes no *SDCARD* do dispositivo atacado. Assim o comando *exploit* é inserido e a máquina do atacante fica esperando ser acessado para que ocorra o ataque, conforme é mostrado na figura 22:

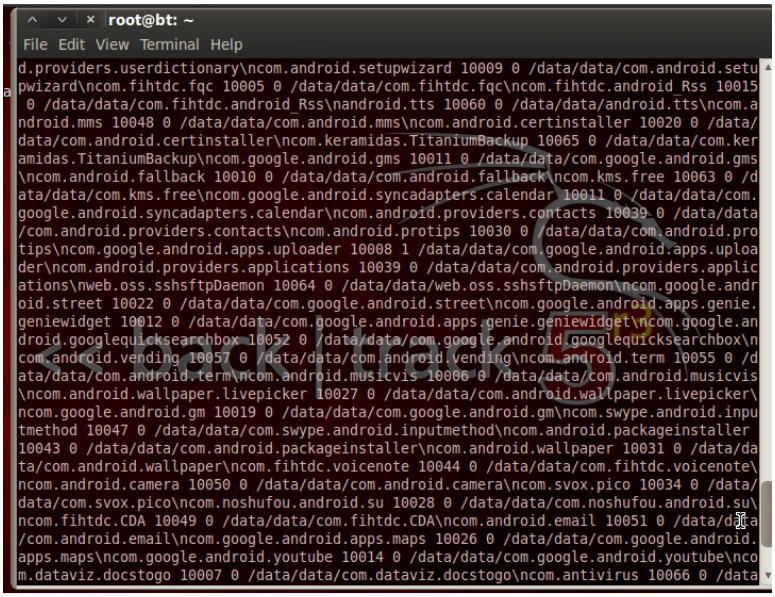
Figura 22 - Máquina do atacante esperando ser acessado



Fonte: da Autora

Após a máquina alvo entrar no *browser* pelo endereço **http://192.168.1.109**, foi iniciado um download automático, onde neste download há um arquivo com extensão html, que ficará armazenada na pasta de downloads do *SDCARD* do dispositivo, neste arquivo, há um código fonte utilizado para acessar todas as pastas do *SDCARD*, e enviar ao atacante informações sobre elas.

Figura 23 - Resultado do ataque realizado



Fonte: da Autora

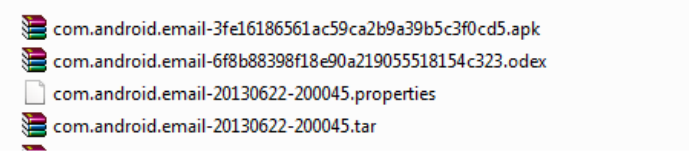
Na figura 23 é possível verificar todas as pastas e arquivos que ali estão armazenados no *SDCARD* do alvo.

7.3.4 Testes Utilizando Aplicativo

Para realizar este ataque, foi utilizado a técnica de engenharia reversa. Segundo Rezende (2005, p.51), ”a reengenharia, também chamada de renovação ou recuperação, não somente recupera informações de um projeto de um software existente, mas usa estas informações para alterar ou reconstituir o sistema existente[...]”. Além de a engenharia reversa ser utilizada para atualizar antigos sistemas, ela também é utilizada por pessoas mal-intencionadas, com a finalidade de inserir códigos maliciosos em aplicativos prontos e disponíveis na loja *Play* do Google. Este tipo de técnica é utilizado para aplicativos pagos na loja *Play* e disponíveis para download em sites maliciosos que disponibilizam estes aplicativos gratuitamente, como visto nos capítulos anteriores.

Para a realização desta técnica foi utilizado um aplicativo disponível na loja *Play* do Google, após foi realizado o *backup* de algum aplicativo instalado no dispositivo, a ferramenta utilizada para realizar este procedimento, foi a *Titanium Backup*, esta ferramenta está disponível para instalação no dispositivo gratuitamente, ela realiza backups de aplicativos e arquivos do *SDCard*. Depois de realizado o *backup* do aplicativo, foi enviado para o computador por via *usb* a pasta onde do backup realizado:

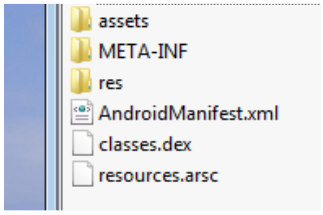
Figura 24 - Backup realizado pela ferramenta Titanium



Fonte: da Autora

Na figura 24, é possível verificar que o *backup* criou vários arquivos, sendo um com extensão .apk, que nada mais é que um executável da plataforma *Android*, como este é um arquivo compactado, ele pode ser extraído facilmente, modificando sua extensão de .apk, para .tar.gz ou .zip, assim ele pôde ser extraído no computador, na figura 25, podemos visualizar os arquivos extraídos:

Figura 25 - Arquivos extraídos do aplicativo



Fonte: da Autora

Para acessar ao comando AndroidManifest.xml, foi necessário a utilização de um *converter*, sendo este chamado de AXMLPrinter2.jar, que foi salvo na pasta onde o aplicativo foi extraído. Em seguida foi digitado o comando no prompt de comando, acessando a pasta do aplicativo extraído e do conversor: **java –jar ./AXMLPrinter2.jar AndroidManifest.xml > AndroidManifestConvertido.xml**, com este comando, foi possível a conversão do arquivo, sendo criado outro arquivo AndroidManifest.xml., também foi utilizado um conversor de classes.dex salva pelo backup para um pacote .jar no formato java, para realizar esta conversão, foi utilizado o comando **dex2jar\dex2jar.bat classes.dex,** após um arquivo é criado com o nome "classes.dex.dex2jar.jar"**.** A última parte para a obtenção de todos os arquivos do aplicativo, foi a conversão do pacote de extensão .jar para código fonte java, para isso foi necessário utilizar um conversor chamado *Java Decompiler*, onde este apenas foi executado e o arquivo "classes.dex.dex2jar.jar" selecionado, assim os arquivos foram convertidos e salvos em um arquivo comprimido, sendo necessário apenas extrair as classes. Na IDE Eclipse, foi criado um novo projeto para os códigos já existentes, as classes foram abertas e o arquivo "adRequest.java" foi modificado, a linha em que contém a versão foi modificada de 4.1.0 para 2.1.0, como pode-se verificar nas figuras 26 e 27:

Figura 26 - Arquivo sem alteração

```
package com.google.ads;

import android.content.Context;

public class AdRequest
{
  public static final String LOGTAG = "Ads";
  public static final String TEST_EMULATOR = ;
  public static final String VERSION = "4.1.0";
  private Gender a = null;
  private String b = null;
  private Set<String> c = null;
  private Map<String, Object> d = null;
  private Location e = null;
  private boolean f = 0;
  private boolean g = 0;
  private Set<String> h = null;
```

Fonte: da Autora

Figura 27 - Arquivo com alteração

```
ackage com.google.ads;

mport android.content.Context;

ublic class AdRequest

  public static final String LOGTAG = "Ads";
  public static final String TEST_EMULATOR = ;
  public static final String VERSION = "2.1.0";
  private Gender a = null;
  private String b = null;
  private Set<String> c = null;
  private Map<String, Object> d = null;
  private Location e = null;
```

Fonte: da Autora

Após a realização das alterações, a aplicação é exportada com o formato *apk* para que seja instalada no dispositivo móvel. A intenção de realizar esta técnica é para averiguar se as ferramentas de segurança possuem bloqueio contra aplicativos sem a assinatura da loja *Play* do *Android*.

7.3.5 Teste realizado com SMS e Ligações Telefônicas

Para a realização deste teste, foi utilizado um dispositivo móvel para realizar o envio de SMS e ligações telefônicas, uma vez que as ferramentas de segurança utilizam bloqueios para dispositivos desconhecidos.

Primeiro no dispositivo onde está a ferramenta de segurança instalada, é configurado para que o número do dispositivo à enviar mensagens e ligações telefônicas, seja bloqueada. Após o dispositivo bloqueado realiza ligações e envia mensagens para o dispositivo com a ferramenta de segurança instalada. A intenção de realizar este teste é saber se a ferramenta de segurança realmente consegue bloquear totalmente o dispositivo ao enviar mensagens e realizar ligações telefônicas.

8 RESULTADOS OBTIDOS COM OS TESTES

A seguir são descritos os resultados obtidos a partir dos testes realizados. Os resultados oportunizam uma melhor comparação entre as ferramentas de segurança listadas anteriormente, sendo que para cada ferramenta foi levado em consideração os serviços oferecidos, e se estes protegem ou bloqueiam os testes aqui realizados. Primeiramente foram avaliados em todas as ferramentas os critérios estipulados anteriormente, em seguida foi realizada uma avaliação dos testes de invasão em cada ferramenta, e por ultimo um comparativo total de todas as ferramentas utilizadas no teste.

8.1 FERRAMENTAS DE ANTIVÍRUS PARA A PLATAFORMA ANDROID

Para a realização dos testes com os antivírus, para cada ferramenta, foram apagadas qualquer alteração anterior com a opção de restauração de padrão de fábrica do dispositivo, para que não ocorra interferências no desempenho do dispositivo.

8.1.1 AVG Antivírus - Free

Tabela 3 - Resultado dos testes - AVG Antivírus Free

Ferramenta de Segurança		**AVG Antivírus - Free**
Versão		3.2.3
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque não detectado
	Ataque via Browser	Ataque não detectado
	Bloqueio SMS e Ligações	Mensagens SMS foram bloqueadas totalmente, porém ligações foram parcialmente bloqueadas, levando a ligação para Caixa do dono do dispositivo móvel.
	Aplicações	Detectado no momento da instalação, executou desinstalação.
Desempenho		RAM execução: 20MB RAM execução com varredura: 36MB Armazenamento: 8,22MB CPU: 0,42% Bateria: 0,16%
Deficiência		-Não detectou alguns ataques; -Telefone parcialmente bloqueado quanto

	a ligações de números bloqueados; -Utiliza muita memoria, tanto no estado de execução, quanto no estado da realização da varredura;	
Pontos Positivos	- Configuração para diferentes idiomas; - Interface amigável; -Suporte para diferentes varreduras; -Agendar Verificação; -Proteção em tempo real; -Detecção de aplicativos indesejáveis; -Suporte para gerenciamento do dispositivo, como fechar aplicativos, monitorar utilização de dados, podendo o dispositivo ser desligado quando atingir certa porcentagem de uso de dados; - Bloqueio de dispositivo á distancia por meio do site da AVG;	
Qualidade de Diagnóstico	Diagnóstico não muito robusto apenas com mensagens, sem uma estatística.	
Tempo de resposta da ferramenta	Tempo varredura sistema	2 min. 36 seg.
	Tempo de varredura arquivos SDCard	11 segundos
	Detecção ataque aplicativo	12 segundos

Fonte: da Autora

8.1.1.1 Imagens da Ferramenta AVG Antivírus Free:

Nas images a seguir, é possível visualizar partes do aplicativo AVG antivírus free instalado no dispositivo móvel, sendo que na figura 28 mostra o menu principal do aplicativo, na figura 29 os sub menus referentes ao botão de proteção do menu principal e na figura 30 o sub menu do botão Antifurto do menu principal. Na figura 31 aparecem opções presentes para proteção, na imagem 32 o sub menu do botão proteção do menu principal e na figura 33 as opções de seleção de pasta para varredura do SDCard. Na figura 34 é exibida a continuação da figura 32, referente ao botão privacidade do menu principal e na figura 35 é demonstrada uma imagem obtida através do site da AVG para localização do dispositivo perdido.

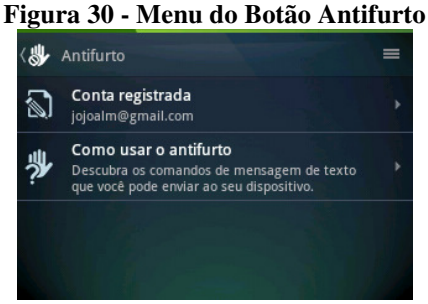
Figura 28 - Menu da Ferramenta AVG

Fonte: da Autora

Figura 29 - Menu do Botão Proteção

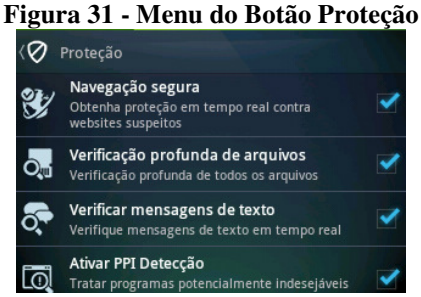
Fonte: da Autora

Figura 30 - Menu do Botão Antifurto



Fonte: da Autora

Figura 31 - Menu do Botão Proteção

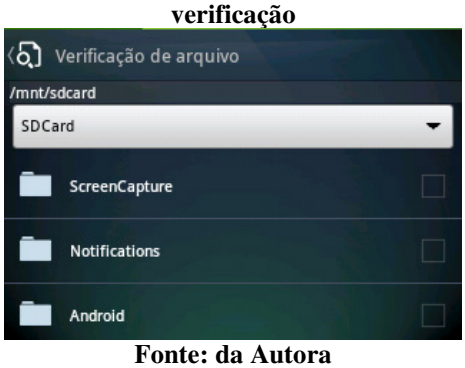


Fonte: da Autora

Figura 32 - Menu do Botão Privacidade

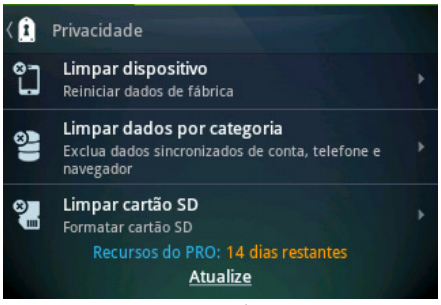
Fonte: da Autora

Figura 33 - Janela para escolha de arquivo para verificação



Fonte: da Autora

Figura 34 - Menu do Botão Privacidade



Fonte: da Autora

Figura 35 - Imagem do site da AVG para bloqueio de dispositivo

Fonte: da Autora

8.1.2 Lookout Security & Antivírus

Tabela 4 - Resultado dos testes - Lookout Security & Antivírus

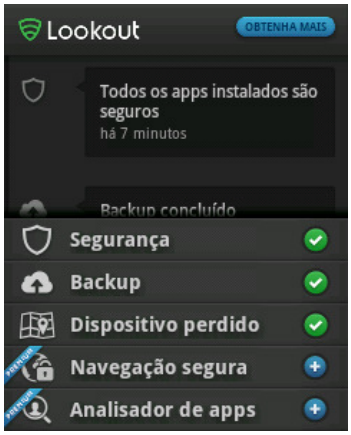
<table>
<tr><th colspan="2">Ferramenta de Segurança</th><th colspan="2">Lookout Security & Antivírus</th></tr>
<tr><td colspan="2">Versão</td><td colspan="2">8.17-8a39d3f</td></tr>
<tr><th colspan="4">Critérios de Teste</th></tr>
<tr><td rowspan="4">Eficiência</td><td colspan="3">Resultados Testes</td></tr>
<tr><td>Ataque Via Bluetooth</td><td colspan="2">Ataque não detectado</td></tr>
<tr><td>Ataque via Browser</td><td colspan="2">Ataque não detectado</td></tr>
<tr><td>Bloqueio SMS e Ligações</td><td colspan="2">Não possui suporte para estes bloqueios</td></tr>
<tr><td>Aplicações</td><td colspan="2">Não detectado</td></tr>
<tr><td colspan="2">Desempenho</td><td colspan="2">RAM execução: 29MB
RAM execução com varredura: 31MB
Armazenamento: 6,18MB
CPU: 9,29%
Bateria: 4.05%</td></tr>
<tr><td colspan="2">Deficiência</td><td colspan="2">-Não detectou nenhum ataque;
-Telefone parcialmente bloqueado quanto á ligações de números bloqueados;
-Utiliza muita memória, tanto no estado de execução, quanto no estado da realização da varredura;
-Muitos recursos são apenas para a versão paga;</td></tr>
<tr><td colspan="2">Pontos Positivos</td><td colspan="2">- Interface amigável;
-Agendar Verificação;
-Proteção em tempo real;
- Localização de dispositivo;</td></tr>
<tr><td colspan="2">Qualidade de Diagnóstico</td><td colspan="2">Diagnóstico não muito robusto apenas com mensagens, sem uma estatística.</td></tr>
<tr><td colspan="2" rowspan="3">Tempo de resposta da ferramenta</td><td>Tempo varredura sistema</td><td>29 seg.</td></tr>
<tr><td>Tempo de varredura arquivos SDCard</td><td>Não possui este suporte</td></tr>
<tr><td>Detecção ataque aplicativo</td><td>Não foi detectado</td></tr>
</table>

Fonte: da Autora

8.1.2.1 Imagens da Ferramenta Lookout Security & Antivírus

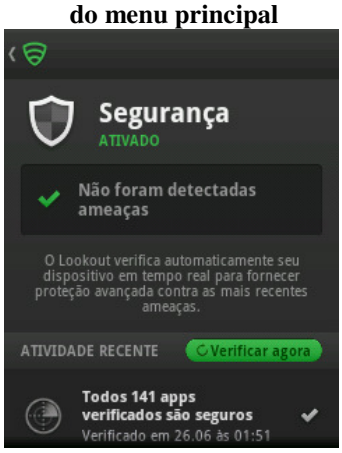
Nas imagens a seguir, podem ser visualizadas partes do aplicativo Lookout Security & Antivírus, instalado no dispositivo móvel. Na figura 36 podemos visualizar o menu principal da ferramenta, na figura 37 podemos visualizar a página de segurança do aplicativo, onde é definido se o aplicativo está ou não habilitado e também um botão para realizar a varredura do dispositivo. Na figura 38 podemos visualizar a página para realização de backup no dispositivo, a figura 40 mostra um mapa para encontrar o dispositivo quando perdido.

Figura 36 - Menu Principal ferramenta Lookout



Fonte: da Autora

Figura 37 – Página do botão de segurança do menu principal



Fonte: da Autora

Figura 38 – Página do Botão Backup

Fonte: da Autora

Figura 39 – Página do Botão Dispositivo Perdido

Fonte: da Autora

Figura 40 – Página de Dispositivo Perdido

Fonte: da Autora

8.1.3 Dr. Web Antivírus Light

Tabela 5 - Resultados dos testes - Dr.Web Antivírus Light

Ferramenta de Segurança		**Dr.Web Antivírus Light**
Versão		7.00.4
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque não detectado
	Ataque via Browser	Ataque não detectado
	Bloqueio SMS e Ligações	Não Possui Suporte para este bloqueio
	Aplicações	Detectado no momento da instalação
Desempenho		RAM execução: 2,8MB RAM execução com varredura: 2,8MB Armazenamento: 1,43MB CPU: 14,17% Bateria: 0.27%
Deficiência		-Não detectou alguns ataques; -Não possui configurações para diferentes idiomas;
Pontos Positivos		- Interface amigável; -Suporte para diferentes tipos de varreduras; -Proteção em tempo real; -Detecção de aplicativos indesejáveis; -Mostra estatísticas das varreduras; -Quarentena para ameaças;
Qualidade de Diagnóstico		Possui diagnóstico bom, informando quantos arquivos forma verificados, ameaças encontradas, especificando são as ameaças.

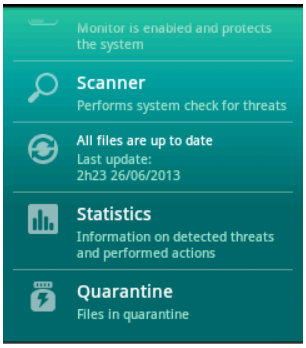
Tempo de resposta da ferramenta	Tempo varredura sistema	12 min. 15 seg.
	Tempo de varredura arquivos SDCard	1 segundo
	Detecção ataque aplicativo	7 segundos

Fonte: da Autora

8.1.3.1 Imagens da ferramenta Dr. Web antivírus Light

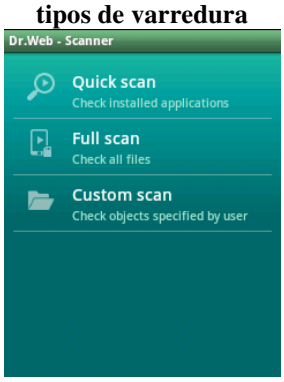
A seguir é possível visualizar as imagens do aplicativo Dr. Web antivírus Light no dispositivo instalado. Na figura 41 podemos visualizar o menu principal da ferramenta com várias opções para uso desta, na figura 42 é possível visualizar a página de varredura do aplicativo, com diversos tipos de varredura. Na figura 43 podemos visualizar opções para seleção de varredura no SDCard, na figura 44 podemos visualizar uma estatística da varredura realizada no dispositivo e a opção de quarentena.

Figura 41 - Menu Inicial da ferramenta Dr. Web antivírus



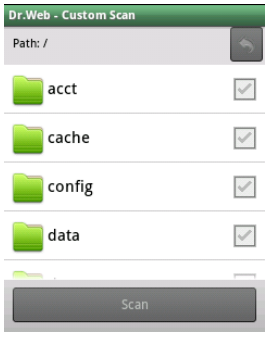
Fonte: da Autora

Figura 42 – Página com todos tipos de varredura



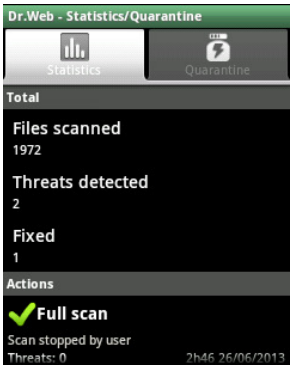
Fonte: da Autora

Figura 43 – Página da opção Custom scan



Fonte: da Autora

Figura 44 – Estatística da ferramenta



Fonte: da Autora

8.1.4 Avast ! Mobile Security

Tabela 6 - Resultados dos testes - Avast! Mobile Security

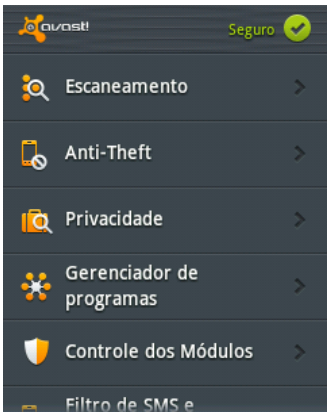
Ferramenta de Segurança		**Avast! Mobile Security**	
Versão		2.0.4993	
Critérios de Teste			
Eficiência		**Resultados Testes**	
	Ataque Via Bluetooth	Ataque não detectado	
	Ataque via Browser	Ataque não detectado	
	Bloqueio SMS e Ligações	Mensagens SMS foram bloqueadas totalmente, porém ligações foram parcialmente bloqueadas, levando a ligação para Caixa do dono do dispositivo móvel.	
	Aplicações	Detectado no momento da instalação	
Desempenho		RAM execução: 11,57MB RAM execução com varredura: 20MB Armazenamento: 5,7MB CPU: 0,63% Bateria: 7,24%	
Deficiência		-Não detectou alguns ataques; -Telefone parcialmente bloqueado quanto á ligações de números bloqueados;	
Pontos Positivos		- Interface muito amigável; -Suporte para diferentes varreduras; -Agendar Verificação; -Proteção em tempo real; -Detecção de aplicativos indesejáveis; -Suporte para gerenciamento do dispositivo, como fechar aplicativos, monitorar utilização de dados, podendo o dispositivo ser desligado quando atingir certa porcentagem de uso de dados;	
Qualidade de Diagnóstico		Possui um log sem muitas informações, somente a quantidade de objetos scaneados, data e horário da varredura.	
Tempo de resposta da ferramenta		Tempo varredura sistema	27 seg.
		Tempo de varredura arquivos SDCard	2 segundos
		Detecção ataque aplicativo	15 segundos

Fonte: da Autora

8.1.4.1 Imagens da ferramenta Avast! Mobile Security

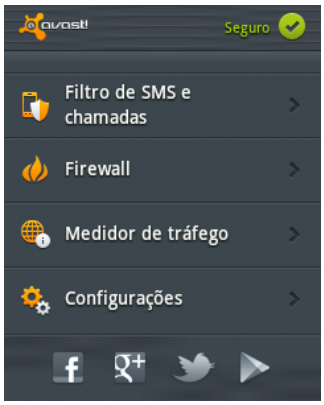
A seguir é possível visualizar imagens do aplicativo Avast! Mobile Security, sendo que nas figuras 46 e 47 é possível visualizar o menu principal do aplicativo, na figura 48 as opções de varredura do aplicativo, na figura 49 as opções de antirroubo. Na figura 50 o gerenciamento de aplicativos, sendo possível fecha-los e na figura 51 opções de privacidade para o usuário.

Figura 45 - Menu Inicial da ferramenta Avast!



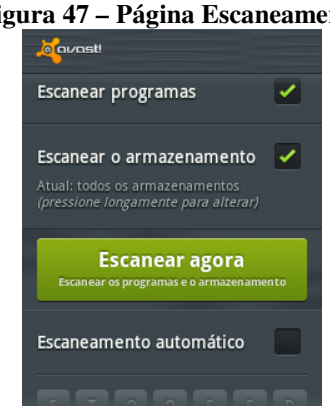
Fonte: da Autora

Figura 46 – Menu Inicial daferramenta Avast!



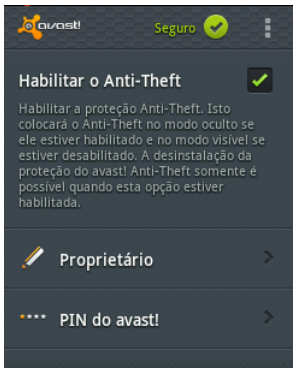
Fonte: da Autora

Figura 47 – Página Escaneamento



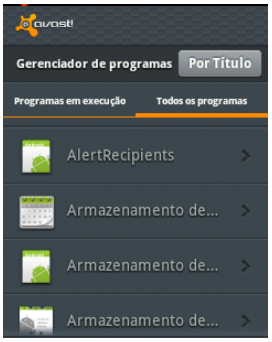
Fonte: da Autora

Figura 48 – Anti-Theft do Avast!



Fonte: da Autora

Figura 49 - Gerenciador de Programas



Fonte: da Autora

Figura 50 – Página de Privacidade

Fonte: da Autora

8.1.5 Norton Mobile Security

Tabela 7 - Resultados testes - Norton Mobile Security

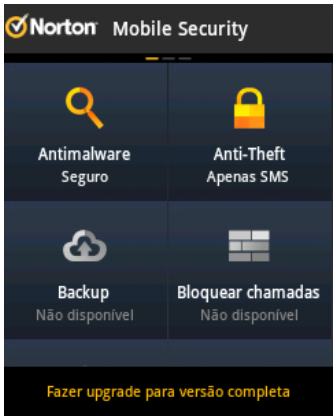
<table>
<tr><th colspan="2">Ferramenta de Segurança</th><th colspan="2">Norton Mobile Security</th></tr>
<tr><td colspan="2">Versão</td><td colspan="2">3.5.0.1023</td></tr>
<tr><th colspan="4">Critérios de Teste</th></tr>
<tr><td rowspan="4">Eficiência</td><td colspan="3">Resultados Testes</td></tr>
<tr><td>Ataque Via Bluetooth</td><td colspan="2">Ataque não detectado</td></tr>
<tr><td>Ataque via Browser</td><td colspan="2">Ataque não detectado</td></tr>
<tr><td>Bloqueio SMS e Ligações</td><td colspan="2">Não Possui Suporte</td></tr>
<tr><td>Aplicações</td><td colspan="2">Não detectado</td></tr>
<tr><td colspan="2">Desempenho</td><td colspan="2">RAM execução: 35MB
RAM execução com varredura: 38MB
Armazenamento: 5,1MB
CPU: 1,9%
Bateria: 1,31%</td></tr>
<tr><td colspan="2">Deficiência</td><td colspan="2">-Não detectou alguns ataques;
-Utiliza muita memória;
-Não possui outros suportes apenas na versão paga;</td></tr>
<tr><td colspan="2">Pontos Positivos</td><td colspan="2">- Interface amigável;
-Agendar Verificação;
-Proteção em tempo real;
-Detecção de aplicativos indesejáveis;
-Proteção Antirroubo;</td></tr>
<tr><td colspan="2">Qualidade de Diagnóstico</td><td colspan="2">Possui um log sem muitas informações, somente a quantidade de objetos scaneados, data e horário da varredura.</td></tr>
<tr><td colspan="2" rowspan="3">Tempo de resposta da ferramenta</td><td>Tempo varredura sistema</td><td>2 min e 30 seg.</td></tr>
<tr><td>Tempo de varredura arquivos SDCard</td><td>Não possui</td></tr>
<tr><td>Detecção ataque aplicativo</td><td>Não detectou nenhum ataque</td></tr>
</table>

Fonte: da Autora

8.1.5.1 Imagens da Ferramenta Norton Mobile Security

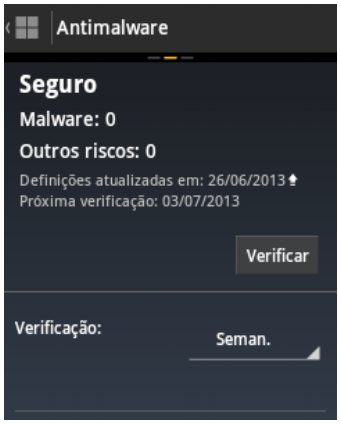
Nas figuras a seguir pode ser visualizado o aplicativo Norton Mobile Security instalado no dispositivo móvel, onde na figura 52 está o menu principal do aplicativo, na figura 53 opção de varredura no dispositivo, na figura 54 opções de antirroubo do aplicativo e na figura 55 algumas opções presente no aplicativo em sua versão paga.

Figura 51 - Menu principal da ferramenta Norton Mobile Security



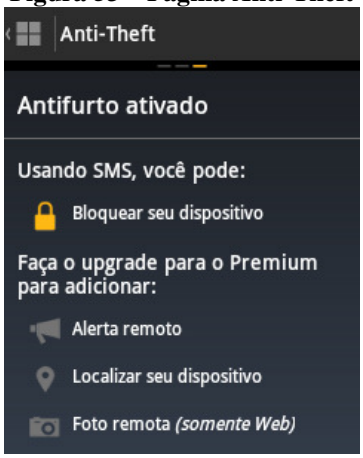
Fonte: da Autora

Figura 52 – Página do botão Antimalware



Fonte: da Autora

Figura 53 – Página Anti-Theft



Fonte: da Autora

Figura 54 –Página com serviços disponíveis na versão paga

Fonte: da Autora

8.1.6 Eset Mobile Security

Tabela 8 - Resultados testes - ESET Mobile Security

Ferramenta de Segurança		**Eset Mobile Security**	
Versão		1.1.995.1221 – 30 dias de licença	
Critérios de Teste			
Eficiência	**Resultados Testes**		
	Ataque Via Bluetooth	Ataque não detectado	
	Ataque via Browser	Ataque não detectado	
	Bloqueio SMS e Ligações	Mensagens SMS foram bloqueadas totalmente, porém ligações foram parcialmente bloqueadas, levando a ligação para Caixa do dono do dispositivo móvel.	
	Aplicações	Detectado	
Desempenho		RAM execução: 13MB RAM execução com varredura: 20MB Armazenamento: 3,44MB CPU: 13,96% Bateria: 0,3%	
Deficiência		-Não detectou alguns ataques;	
Pontos Positivos		- Interface amigável; -Agendar Verificação; -Proteção em tempo real; -Detecção de aplicativos indesejáveis; -Proteção Antirroubo; -Suporte para outros idiomas; -Tipos diferentes de varredura;	
Qualidade de Diagnóstico		Cria logs com informações bem definidas	
Tempo de resposta da ferramenta		Tempo varredura sistema	4 min. 57seg.
		Tempo de varredura arquivos SDCard	1segundo
		Detecção ataque aplicativo	13 segundos

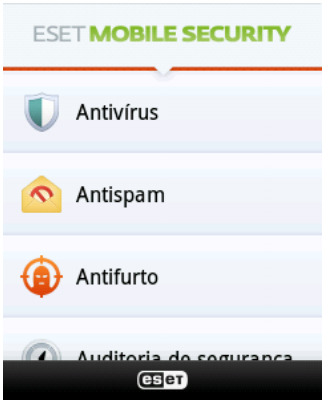
Fonte: da Autora

8.1.6.1 Imagens da Ferramenta Eset Mobile Security

Nas figuras a seguir pode ser visualizado o aplicativo Eset Mobile Security, instalado no dispositivo, onde nas figuras 56 e 57 é possível visualizar o menu principal do aplicativo com opções para sua utilização, na figura 58 opções para configurar

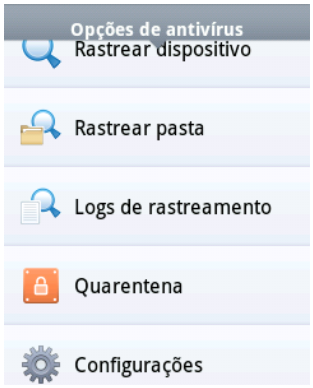
AntiSpam e ligações no dispositivo, na figura 59 opções de configuração para antifurto e na figura 60 opções para gerenciamento de aplicações instaladas no dispositivo, podendo monitorá-las a partir desta página.

Figura 55 - Menu principal da ferramenta Eset



Fonte: da Autora

Figura 56 – Menu Inicial da ferramenta Eset



Fonte: da Autora

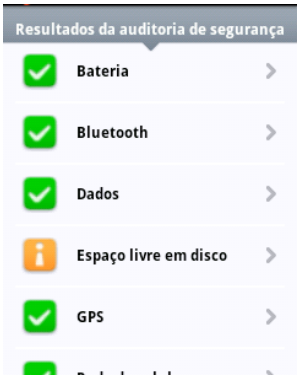
Figura 57 – Página AntiSpam Eset

Fonte: da Autora

Figura 58 – Antifurto Eset

Fonte: da Autora

Figura 59 - Gerenciador eset



Fonte: da Autora

8.2 FERRAMENTAS DE *FIREWALL* PARA A PLATAFORMA ANDROID

Para a realização dos testes com as ferramentas de segurança com *Firewall*, foi necessário inicialmente a liberação do superusuário *ROOT* para o funcionamento do *firewall* no dispositivo, este processo é conhecido como *Rooting*, segundo Braga, Nascimento, Palma e Rosa (2012, p.62),

> Foi criado o termo roooting para se referir ao ganho de acesso irrestrito à plataforma dos dispositivos rodando Android [...] O processo de rooting muda significantemente de dispositivo para dispositivo.[...]Como o Android é derivado do Linux, fazer rooting equivale à obter permissões de acesso administrativo no dispositivo, ou seja, as permissões da conta root. As motivações para a habilitação de tal acesso são várias, como por exemplo: Instalação de versões modificadas do Android; Uso de temas personalizados; Executar modificações no Kernel; Backup de todos os dados, pois é necessário acesso administrativo para se obter tais dados; Ativar funcionalidades que foram bloqueadas por operadoras.

Ainda com processo de *rooting* realizado, o dispositivo utilizado para os testes, não possuía suporte para a utilização do *firewall*. Além do processo de *rooting*, ainda foi necessária a utilização de uma versão modificada do *Android*, uma vez que a versão utilizada pelo dispositivo que era padrão de fabrica, não possuía suporte para a utilização do *firewall*, sendo que quando o *firewall* era instalado apresentava uma mensagem de erro, esta modificação é nomeada como "*Custom ROM*", segundo Cardoso (2012, p.10),

> ROM's customizadas nada mais são do que uma versão personalizada de alguma versão do Android. Estas customizações podem ser construídas a partir das versões que vieram dos fabricantes como podem ser construídas a partir do zero, com base no código do Android disponibilizado pela Google no AOSP(Android Open Soure Project).

A versão utilizada para os testes com as ferramentas de *firewall*, foi a *Remix*2.5, que possui uma modificação capaz de fazer o firewall ser habilitado no dispositivo utilizado para a realização destes testes.

8.2.1 DroidWall

Para a utilização de regras de *firewall* o DroidWall possui uma estrutura própria para a criação delas. A regra criada **$Iptables -A "droidwall" - -destination "192.168.1.0/24" –j "droidwall-reject"**, permite que todas as conexões da rede sejam rejeitadas, **"droidwall"** é utilizado quando as regras a serem criadas passam por alguma interface, além do "droidwall". O aplicativo ainda utiliza outras duas especificações para a criação das regras, "droidwall-3g", esta especificação é utilizada para criar regras quanto aos pacotes de saída podendo ser 3G, 2G, 4G, "droidwall-wifi", esta especificação é utilizada para criar regras para a saída de pacotes da interface wi-fi do dispositivo. A especificação "droidwall-reject" é aplicada quando algum pacote ou acesso deve ser bloqueado pela regra criada. Além da utilização de especificações próprias para a criação das regras, as outras especificações são utilizadas nas regras comuns de iptables, com –A, para inserir uma nova regra para o firewall, -- destination, especificando o destino dos pacotes, no caso da regra criada a rede, e –j onde é informado a ação que será realizada pela regra criada, como neste caso o de rejeitar.

Tabela 9 - Resultados testes - DroidWall

Ferramenta de Segurança		**DroidWall**
Versão		1.5.7
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque Não Bloqueado
	Ataque via Wi-fi	Ataque Bloqueado
Desempenho		RAM execução: 6,26MB Armazenamento: 1,06MB CPU: 0,4% Uso Bateria: 0,03%
Deficiência		-Suporte para diferentes línguas.
Pontos Positivos		- Bloqueio de ataques mal-intencionados na rede; -Bloqueio de acesso para um aplicativo; -Criação de regras customizadas; -Bloqueio para aplicações com acesso á root; -Bloqueio para aplicações com acesso ao kernel;

Fonte: da Autora

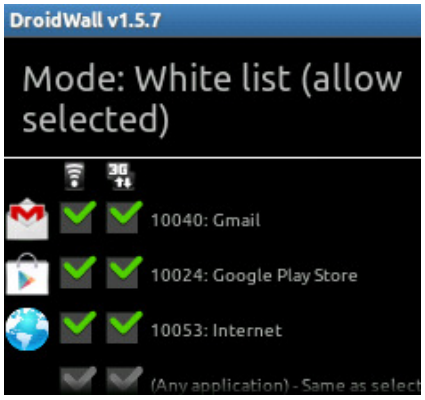
8.2.1.1 Imagens da Ferramenta DroidWall

Nas imagens a seguir, pode ser visualizada a ferramenta de firewall DroidWall instalada no dispositivo, sendo que na figura 61 mostra opções de listas do aplicativo, com White list e black list. Na figura 62 são demonstradas as opções de bloqueios de aplicativos, estas sendo apresentadas em uma lista, na figura 63 o menu principal da ferramenta, podendo habilitá-la ou desabilitá-la e na figura 64 a opção para criar regras de firewall customizadas.

Figura 60 - Modo de escolha da lista Droidwall

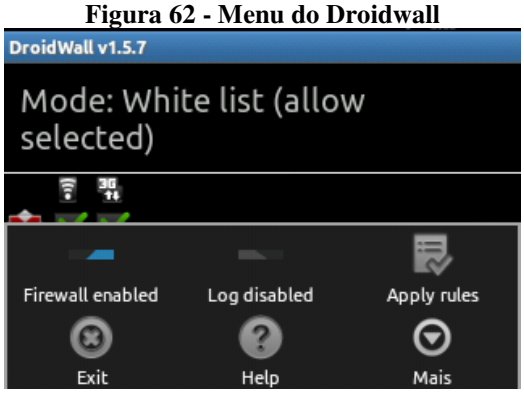
Fonte: da Autora

Figura 61 - Seleção dos aplicativos bloqueados Droidwall



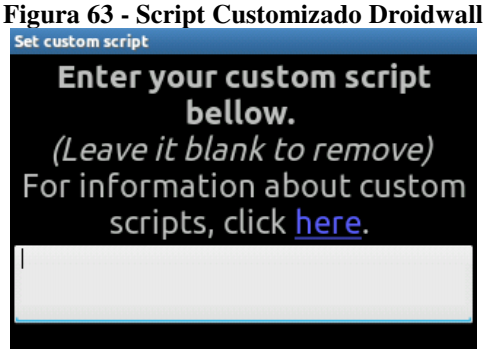
Fonte: da Autora

Figura 62 - Menu do Droidwall



Fonte: da Autora

Figura 63 - Script Customizado Droidwall



Fonte: da Autora

8.2.2 Root Firewall

Para realizar os testes com a ferramenta *Root Firewall*, foi necessário tornar o firewall ativo, e em seguida bloquear os acessos via rede dos serviços do *browser* e dos serviços de *bluetooth*, assim foi possível averiguar se o dispositivo poderia bloquear o acesso dos ataques. Também foi testada a eficiência do *firewall* em bloquear aplicativos quanto a seu acesso na rede, também de bloquear acesso à rede de aplicações com acesso ao *root* e ao *kernel*. Este *Firewall* não possui a possibilidade de criar regras especificas, assim só foi possível testar bloqueando os serviços. O *firewall* possui duas opções de listas para bloqueio, uma de lista negra, onde o aplicativo a ser bloqueado deverá ser selecionado e os outros aplicativos não selecionados são liberados, e o de lista branca, onde os aplicativos selecionados são liberados para utilizar a rede e os não selecionados são bloqueados.

Tabela 10 - Resultados testes - Root Firewall

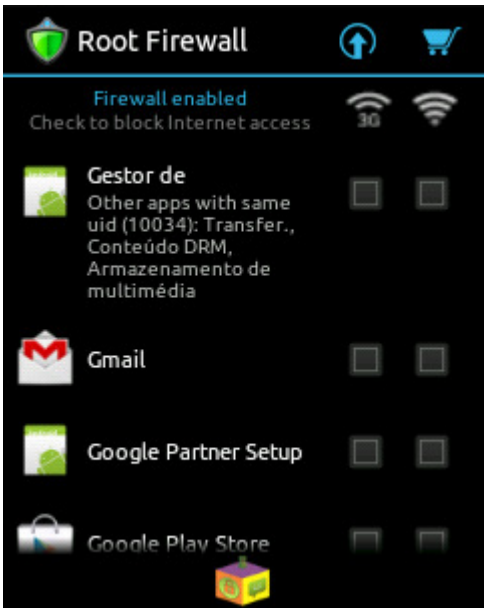
Ferramenta de Segurança		**Root Firewall**
Versão		1.2
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque Não Bloqueado
	Ataque via Wi-fi	Ataque Bloqueado
Desempenho		RAM execução: 13,49MB Armazenamento: 1,11MB CPU: 0,65% Uso Bateria: 0,09%
Deficiência		-Suporte para diferentes línguas. -Criação de regras customizadas.
Pontos Positivos		- Bloqueio de ataques mal-intencionados á rede; -Bloqueio de acesso para um aplicativo; -Bloqueio para aplicações com acesso á root; -Bloqueio para aplicações com acesso ao kernel;

Fonte: da Autora

8.2.2.1 Imagens da Ferramenta *Root Firewall*

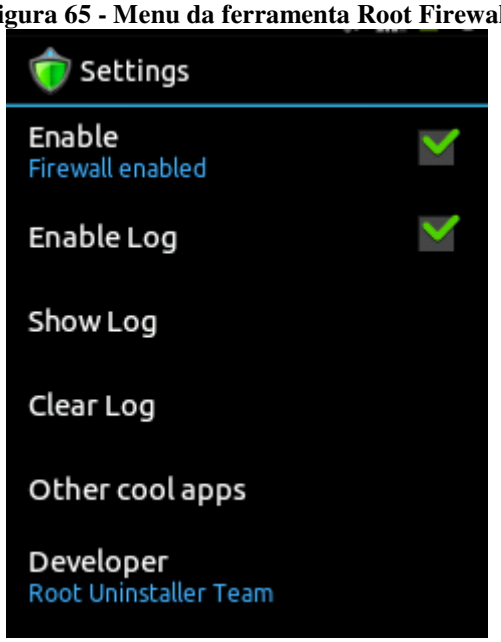
A seguir é possível visualizar figuras da ferramenta *Root Firewall* instalada no dispositivo móvel, na figura 65 pode se visualizar as opções de boqueio para aplicativos, onde estas são apresentadas em uma lista e na figura 66 a opção do menu principal do aplicativo, permitindo habilitá-la ou desabilitá-la.

Figura 64 - Bloqueio de aplicativos Root Firewall



Fonte: da Autora

Figura 65 - Menu da ferramenta Root Firewall



Fonte: da Autora

8.2.3 Bluetooth Firewall Trial

A ferramenta *Bluetooth Firewall Trial*, é uma ferramenta que possui como foco o bloqueio e proteção apenas do *bluetooth* do dispositivo, possuindo vários serviços de alerta, como alertar se o *bluetooth* está ativo, alertar se o dispositivo estiver em modo de descoberta, alerta se o dispositivo teve o nome modificado e avisar se um outro dispositivo conectou. Esta ferramenta não possui a possibilidade de criar regras especificas.

Tabela 11 - Resultados testes - Bluetooth Firewall Trial

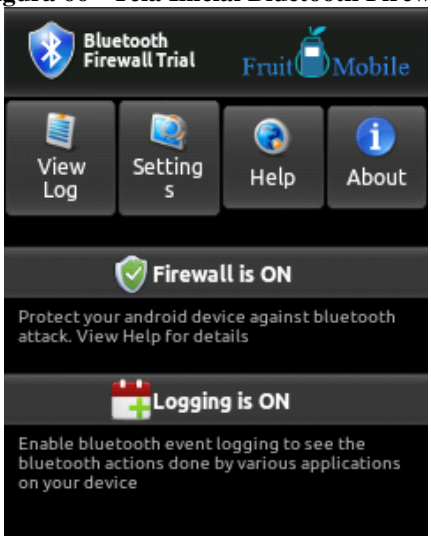
Ferramenta de Segurança		**Bluetooth Firewall Trial**
Versão		1.0
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque Bloqueado
	Ataque via Wi-fi	Não Bloqueado
Desempenho		RAM execução: 10,25MB Armazenamento: 745KB CPU: 11,12% Uso Bateria: 0,0%
Deficiência		-Suporte para diferentes línguas. -Criação de regras customizadas. -Bloqueio de acesso para um aplicativo; -Bloqueio para aplicações com acesso á root; -Bloqueio para aplicações com acesso ao kernel;
Pontos Positivos		- Bloqueio de ataques mal-intencionados ao bluetooth;

Fonte: da Autora

8.2.3.1 Imagens da Ferramenta *Bluetooth Firewall Trial*

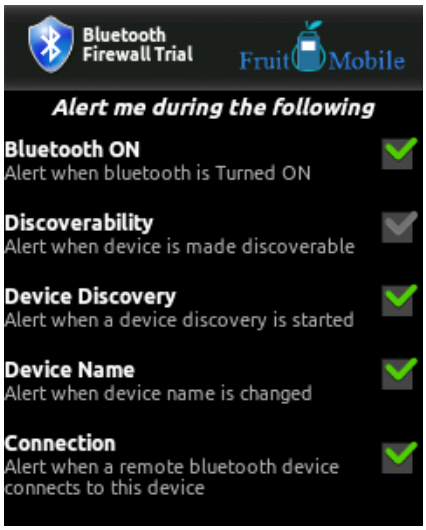
A seguir podem ser visualizadas as figuras do aplicativo instalado no dispositivo móvel, na figura 67 podemos visualizar o menu principal da ferramenta, podendo habilitá-la ou desabilitá-la, na figura 68 podemos visualizar opções de alerta do bluetooth e na figura 69 um sub menu com algumas opções de configuração do aplicativo.

Figura 66 - Tela Inicial Bluetooth Firewall



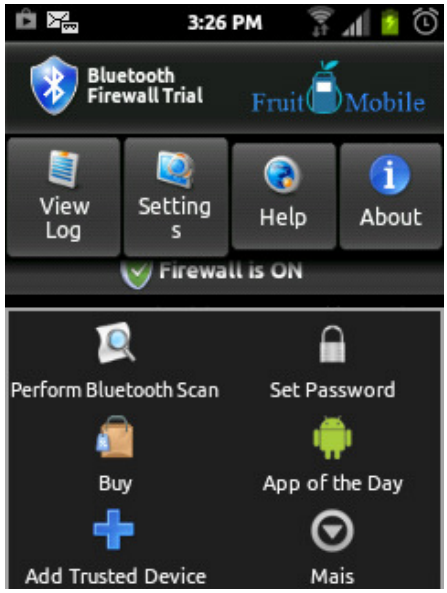
Fonte: da Autora

Figura 67 - Menu da ferramenta Bluetooth Firewall



Fonte: da Autora

Figura 68 - Menu ferramenta Bluetooth Firewall



Fonte: da Autora

8.2.4 Avast *Mobile Security* & Antivirus

A ferramenta Avast, possui integrado ao seu antivírus, o serviço de *firewall*, onde este possui o funcionamento de bloqueio de aplicativos e serviços do dispositivo à rede, podendo escolher tanto a wi-fi ou a 2g/3g, ou as duas. Possui possibilidade de criar listas de bloqueio, tanto lista branca quanto lista negra, onde quando utilizado a lista negra, o aplicativo a ser bloqueado deverá ser selecionado e os outros aplicativos não selecionados são liberados, e o de lista branca, onde os aplicativos selecionados são liberados para utilizar a rede e os não selecionados são bloqueados. O Avast também possibilita a criação de regras especificas, porém não em forma de script, mas sim com um *template* já pronto, sendo a regra a ser criada é pré–definida. Como é possível ver na figura 72, os campos a serem preenchidos, são o nome da regra, se ela estará ativada, o tipo de conexão, se a regra irá bloquear ou permitir acesso, o tipo de regra se utilizara IP ou porta ou os dois, sendo possível inserir a porta e o endereço IP manualmente.

Tabela 11 - Resultados dos testes - Avast Mobile Security & Antivírus- Firewall

Ferramenta de Segurança		**Avast Mobile Security & Antivírus**
Versão		2.0.4993
Critérios de Teste		
Eficiência	**Resultados Testes**	
	Ataque Via Bluetooth	Ataque Não Bloqueado
	Ataque via Wi-fi	Ataque Bloqueado
Desempenho		RAM execução: 11,53MB Armazenamento: 5,70MB CPU: 2,73% Uso Bateria: 0,17%
Deficiência		-Nenhuma
Pontos Positivos		-Suporte para diferentes línguas. - Bloqueio de ataques mal-intencionados á rede; -Bloqueio de acesso para um aplicativo; -Criação de regras customizadas; -Bloqueio para aplicações com acesso á root; -Bloqueio para aplicações com acesso ao kernel;

Fonte: da Autora

8.2.4.1 Imagens da Ferramenta Avast!

A seguir podem ser visualizadas algumas imagens do aplicativo Avast! na opção de firewall presente na ferramenta. Na figura 70 estão as opções de bloqueio para aplicativos instalados no dispositivo e na figura 71 mostra a opção que a ferramenta apresenta para criação de regras de firewall customizados.

Figura 69 - Modo de Bloqueio por aplicativo Avast!

Fonte: da Autora

Figura 70 - Regra customizada Avast!

Fonte: da Autora

8.3 FERRAMENTA VPN

As ferramentas de VPN para a Plataforma *Android* possuem uma forma diferente de utilizar esta rede criptografada, normalmente conectando-se a servidores externos, podendo ser até de outros países, também permitem o acesso a sites bloqueados. A seguir serão testadas algumas das ferramentas de VPN disponíveis para *Android*.

Para os testes com as ferramentas VPN, foi considerado o desempenho de cada ferramenta, também os pontos positivos e deficiências. Quanto aos testes de invasão, não foi possível a realização, uma vez que as redes criadas pelos serviços de VPN, utilizam conexões com servidores remotos, sendo quase impossível conseguir esta conexão, pois estas não conectavam ao dispositivo.

8.3.1 VPN One Click

Esta ferramenta possibilita a criação de uma rede VPN a partir de um servidor remoto, tendo como opções vários países, como EUA, Alemanha, Canadá, Austrália, dentre outros países no mundo, porém este serviço é disponibilizado apenas para usuários com conta Elite. Para utilizar o serviço gratuito, é oferecida a opção de conectar a um servidor de algum país aleatoriamente. A conexão com um servidor remoto não foi realizada, uma vez que o aplicativo tenta conectar e apenas processa o pedido, não obtendo nenhuma conexão.

Tabela 12 -Resultados dos testes - VPN One Click

Ferramenta de Segurança	**VPN One Click**
Versão	Indefinido
Critérios de Teste	
Desempenho	RAM execução: 11,65MB Armazenamento: 1,03MB CPU: 1,22% Uso Bateria: 6,73%
Deficiência	-Não realizou conexão com servidor remoto.
Pontos Positivos	-Ocultação de endereço IP; -Conexão Internet segura;

Fonte: da Autora

8.3.1.1 Imagem da Ferramenta VPN One Click

Na figura 72 podem ser visualizadas as opções de conexão da ferramenta VPN One Click, sendo listadas várias opções de conexão em diferentes países.

Figura 71 - Imagem da tela da ferramenta VPN One Click

Fonte: da Autora

8.3.2 *Hideman* VPN

A ferramenta *Hideman* VPN, disponibiliza seus serviços gratuitamente somete 5 horas por semana, oferecendo recurso de conexão a um servidor remoto para utilização de uma rede VPN, porém sua conexão não é realizada com sucesso, uma vez que a conexão é realizada através de uma requisição por parte do aplicativo, porém a conexão não é estabelecida.

Tabela 12 - Resultados dos testes Hideman VPN

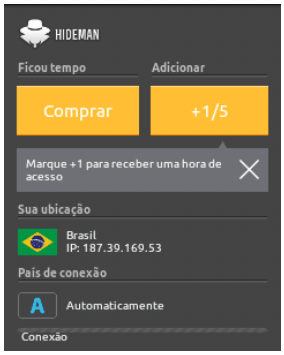
Ferramenta de Segurança	**Hideman VPN**
Versão	2.1
Critérios de Teste	
Desempenho	RAM execução: 3.91MB Armazenamento: 1,67MB CPU: 9,41% Uso Bateria: 42,35%
Deficiência	-Conexão não estabelecida
Pontos Positivos	-Ocultação de endereço IP; -Conexão Internet segura;

Fonte: da Autora

8.3.2.1 Imagens da Ferramenta *Hideman* VPN

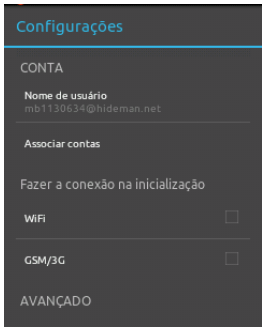
Nas figuras a seguir é possível visualizar a ferramenta de VPN Hideman VPN, na figura 73 é apresentada a página principal da ferramenta Hideman VPN, da figura 74 mostra opçoes de configuração da ferramenta e na figura 75 as opções de conexão de redes privadas em diferentes países.

Figura 72 - Inicio Hideman VPN



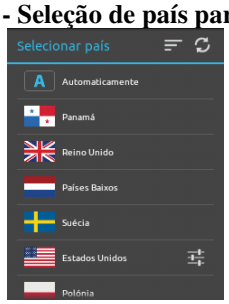
Fonte: da Autora

Figura 73 - Configurações da ferramenta Hideman VPN



Fonte: da Autora

Figura 74 - Seleção de país para conexão



Fonte: da Autora

8.3.3 Viatun 4 VPN

A ferramenta Viatun 4 VPN, possui uma interface diferente das outras ferramentas de VPN, porém o serviço de conexão da VPN com servidores remotos é similar, pode-se criar a rede VPN remotamente modificando a opção *OFF* para *ON*, sua conexão não foi bem realizada, resultando com uma mensagem de falha ao conectar.

Tabela 13 - Resultados dos testes - Viatun 4 VPN

Ferramenta de Segurança	**Viatun 4 VPN**
Versão	6.0
Critérios de Teste	
Desempenho	RAM execução: 11,65MB Armazenamento: 1,03MB CPU: 1,22% Uso Bateria: 6,73%
Deficiência	-Conexão não realizada
Pontos Positivos	-Ocultação de endereço IP; -Conexão Internet segura;

Fonte: da Autora

8.3.3.1 Imagem da Ferramenta Viatun 4 VPN

Na figura 76 pode-se visualizar a ferramenta Viatun 4 VPN instalada no dispositivo, este mostra opção de conectar ou não a uma rede privada, sendo mostrada uma mensagem de *disconnect* quando a rede desconectada e *connect* quando conectada.

Figura 75 - Página Viatun

Fonte: da Autora

9 COMPARAÇÃO ENTRE AS FERRAMENTAS DE SEGURANÇA

As comparações das ferramentas de segurança testadas visam selecionar as ferramentas que obtiveram melhor desempenho. A comparação se dará por meio de gráficos comparativos, conforme segue.

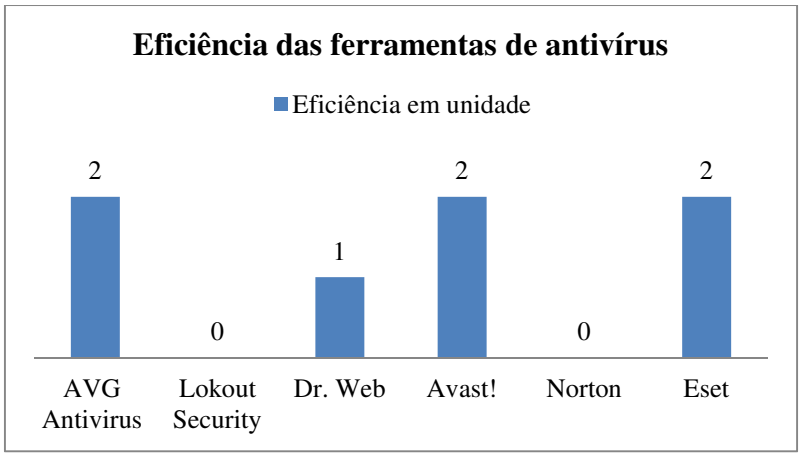
9.1 COMPARAÇÃO ENTRE AS FERRAMENTAS DE ANTIVÍRUS

A seguir é demonstrada através de gráficos uma comparação dos critérios usados nos testes de avaliação das ferramentas de segurança de antivírus, sendo primeiro uma comparação no critério de eficiência, após no critério de desempenho e por último o critério de tempo de resposta do antivírus.

9.1.1 Comparativo das ferramentas de antivírus no critério eficiência

Na tabela abaixo pode-se verificar um comparativo entre as ferramentas de antivírus, quanto a sua eficiência em detectar um ataque realizado, sendo especificada a quantidade em unidade, sendo que um aplicativo detectou um ataque, outros detectaram dois e alguns nenhum.

Gráfico 1 - Eficiência das ferramentas de antivírus



Fonte: da Autora

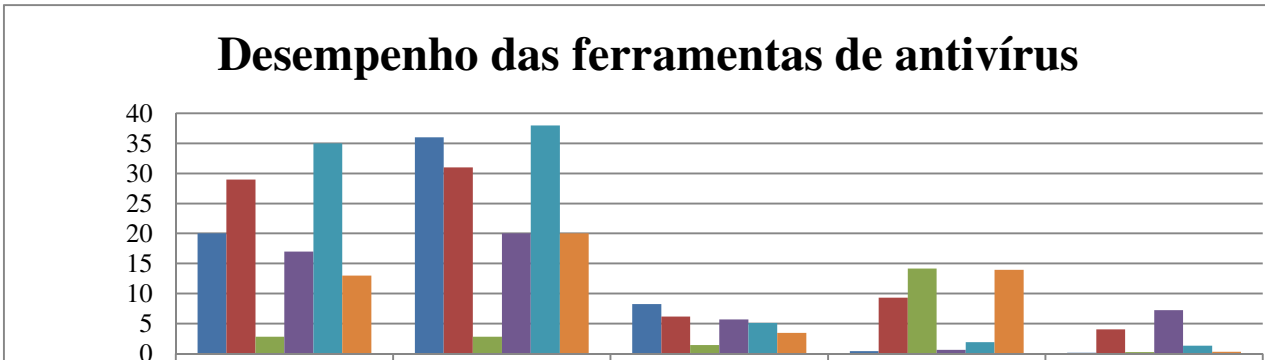
No critério eficiência, podemos verificar pelo gráfico acima, que as ferramentas AVG Antivírus, Avast e Eset, possuíram os melhores resultados, sendo que cada uma conseguiu detectar ou bloquear dois itens do critério eficiência, o antivírus *Dr.Web*

ficou em terceiro conseguindo bloquear um item e os antivírus *Lokout* e Norton bloquearam nenhum item.

9.1.2 Comparativo das ferramentas de antivírus no critério desempenho

Pode ser verificada na tabela abaixo, uma comparação entre as ferramentas de segurança quanto ao desempenho, sendo os seguintes critérios comparados: a de utilização de memória RAM em execução obtida em MB, a utilização de memória RAM com o aplicativo em modo de varredura obtida em MB, armazenamento da ferramenta no dispositivo obtida em MB, o consumo da CPU do dispositivo obtida em porcentagem e o consumo de bateria no dispositivo definida em porcentagem.

Gráfico 2 - Desempenho das ferramentas de antivírus



	RAM Execução(em MB)	RAM com Varredura (em MB)	Armazenamento (em MB)	CPU (consumo em %)	Bateria (consumo em %)
AVG	20	36	8,22	0,42	0,16
Lookout	29	31	6,18	9,29	4,05
Dr. Web	2,8	2,8	1,43	14,17	0,27
Avast!	17	20	5,7	0,63	7,24
Norton	35	38	5,1	1,9	1,31
Eset	13	20	3,44	13,96	0,3

Fonte: da Autora

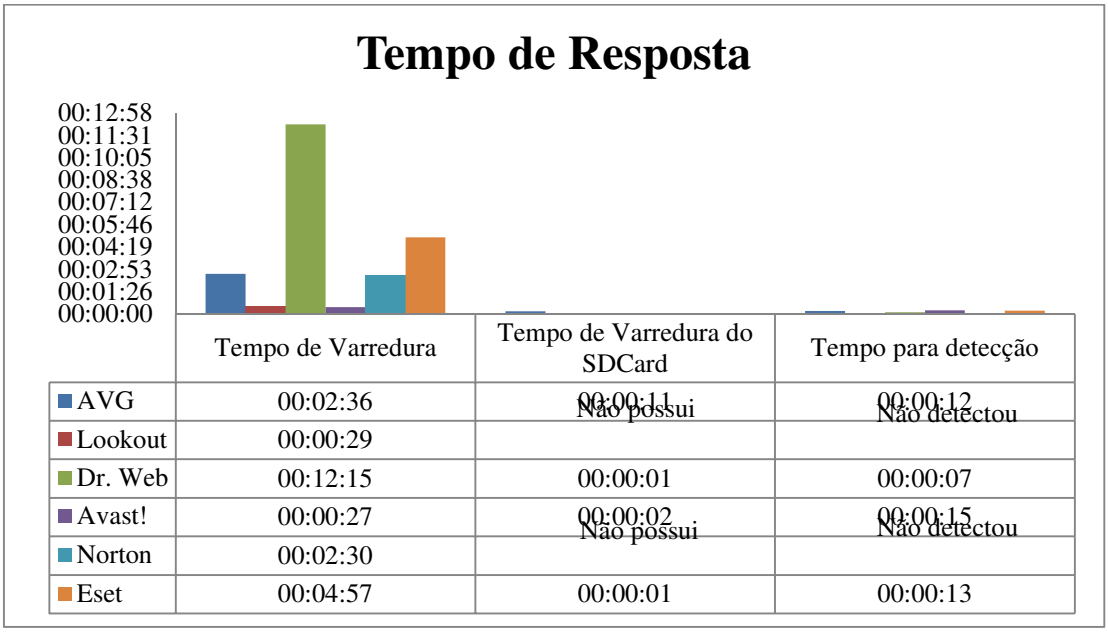
Pela comparação apresentada no gráfico acima, no quesito de utilização de memória RAM do dispositivo que a ferramenta Norton antivírus utiliza uma quantidade maior do que as outras ferramentas com 35MB e a ferramenta *Dr. Web* antivírus, sendo a que menos utilizou memória do dispositivo, com 2,8 RAM. Quando executada a varredura pelo dispositivo, podemos verificar que a ferramenta AVG antivírus, utiliza mais memória RAM do dispositivo, com 36MB e a ferramenta a menos utilizar é a

ferramenta *Dr. Web* antivírus. Na parte de armazenamento do dispositivo, podemos averiguar que a ferramenta *Dr. Web* antivírus, utiliza menos de memória para armazenamento com 1,43 MB e a ferramenta AVG com 8,22MB é a que mais utiliza memória para armazenamento. Do uso da CPU do dispositivo, a ferramenta a menos utilizar é a AVG antivírus, utilizando 0,42% da capacidade de processamento e a ferramenta *Dr.Web* antivírus é a que mais utiliza a CPU, com 14,17%. Quanto à bateria, a ferramenta que mais consome energia é o Avast! com 7,24% e a que menos consome é o Eset antivírus utilizando apenas 0,3% da bateria.

9.1.3 Comparativo das ferramentas de antivírus no critério tempo de resposta

Para a comparação do critério de tempo de resposta da ferramenta no dispositivo, foi utilizado o tempo em que os aplicativos levaram para realizar uma determinada tarefa a eles demandada, como: o tempo de varredura da ferramenta em todo o dispositivo, o tempo em que a ferramenta demorou em realizar a varredura no SDCard do dispositivo e o tempo em que a ferramenta demorou em detectar um ataque no dispositivo. Os resultados são mostrados no formato de horas, minutos e segundos, sendo que na tabela abaixo nos campos “Não possui”, significa que a ferramenta não possuía este atributo, e nos campos “Não detectou”, significa que a ferramenta não conseguiu realizar nenhuma detecção de ataque.

Gráfico 23- Tempo de resposta das ferramentas de antivírus



	Tempo de Varredura	Tempo de Varredura do SDCard	Tempo para detecção
AVG	00:02:36	Não possui	Não detectou
Lookout	00:00:29		
Dr. Web	00:12:15	00:00:01	00:00:07
Avast!	00:00:27	00:00:02	00:00:15
Norton	00:02:30	Não possui	Não detectou
Eset	00:04:57	00:00:01	00:00:13

Fonte: da Autora

Ao verificar o tempo de resposta das ferramentas de antivírus, podemos averiguar que no item de varredura em todo o dispositivo, a ferramenta *Dr.Web* antivírus, foi a mais demorada em realizar a varredura, com 12 minutos, a ferramenta que utilizou menos tempo, foi o Avast, gastando 27 segundos. No quesito de varredura do *SDCard*, a ferramenta que utilizou mais tempo, foi a AVG, com 11segundos e as ferramentas que utilizaram menos tempo foram as *Dr.web* e ESET com 1 segundo. Quanto ao tempo para realizar a detecção de uma ameaça, a ferramenta que detectou mais rapidamente, foi a ferramenta *Dr. Web* utilizando 7 segundos e a que mais demorou foi a ferramenta Avast com 15 segundos, porem as ferramentas *Lookout* e Norton não detectaram a intrusão e não possuem o item de varredura do *SDCard*.

9.1.4 Avaliação das melhores ferramentas de antivírus

Pelos resultados obtidos nos testes realizados, foi possível averiguar que as ferramentas com melhor custo beneficio, foram a ferramenta de antivírus ESET, devido a sua eficiência na detecção dos ataques, por utilizar menos recursos do dispositivo e também por possuir vários serviços, como suporte para diferentes idiomas, antifurto, bloqueio de telefones indesejáveis, além de apresentar um log de verificação completa. Também está incluso nesta lista o Avast!, uma vez que obteve uma boa eficiência na detecção dos ataques, e também não utilizou muito os recursos do dispositivo como processamento da CPU e quantidade de memória RAM do dispositivo, além de disponibilizar ao usuário outros serviços, como o de antifurto, bloqueio de telefones indesejáveis, e também por possuir firewall integrado em seu antivírus.

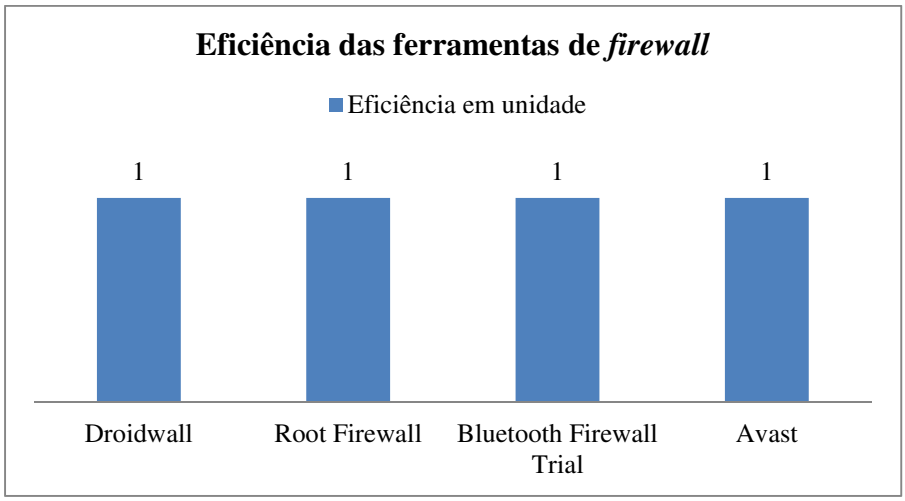
9.2 COMPARAÇÃO ENTRE AS FERRAMENTAS DE *FIREWALL*

A seguir é demonstrada através de gráficos uma comparação dos critérios usados nos testes de avaliação das ferramentas de segurança de *firewall*, sendo primeiro uma comparação no critério de eficiência e após no critério de desempenho.

9.2.1 Comparativo das ferramentas de firewall no critério eficiência

No gráfico abaixo são apresentado os resultados obtidos quanto a eficiência das ferramentas de firewall no bloqueio dos ataques realizados, sendo que cada ferramenta obteve o bloqueio de um ataque.

Gráfico 3 - Eficiência das ferramentas de firewall



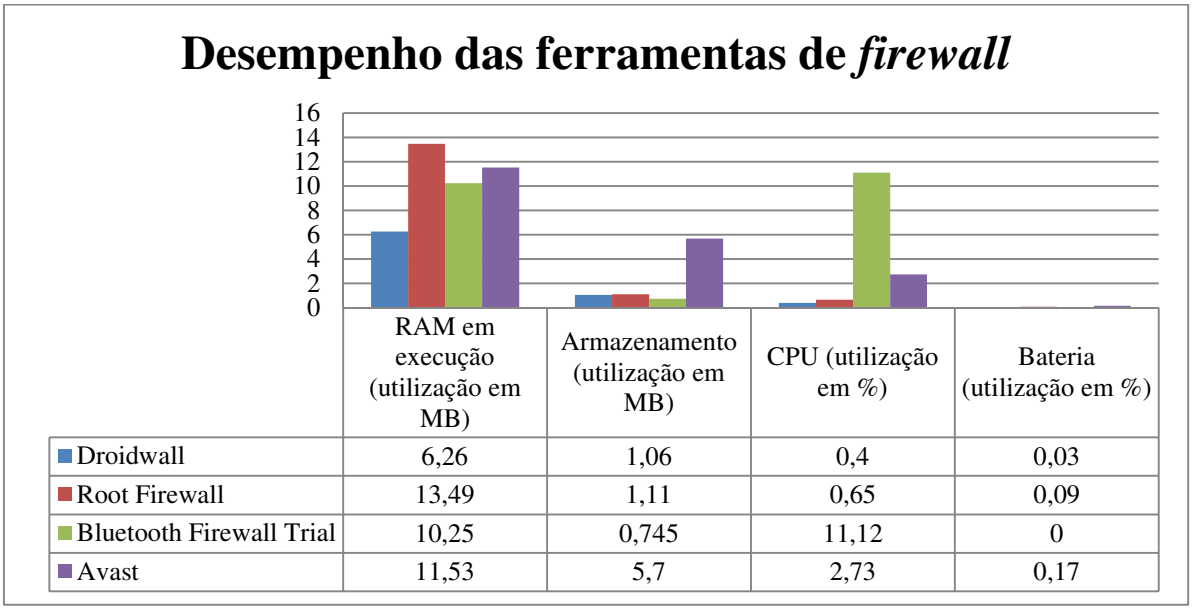
Fonte: da Autora

Conforme podemos visualizar na tabela acima, no critério de eficiência, as ferramentas de *firewall* DroidWall, *Root Firewall* e Avast conseguiram bloquear um ataque, sendo que para a proteção que o *firewall* disponibiliza em todas as ferramentas testadas, é a de bloqueio a redes *wi-fi* e 2g/3g, enquanto a ferramenta *Bluetooth Firewall Trial*, possui bloqueio apenas para *bluetooth* e não para redes *wi-fi* e 2g/3g.

9.2.2 Comparativo das ferramentas de *firewall* no critério desempenho

No gráfico a seguir é demonstrado o comparativo das ferramentas no critério de desempenho, sendo testadas a utilização da memória RAM em execução em MB, a utilização de armazenamento da ferramenta no dispositivo apresentada em MB e a utilização da CPU e bateria em porcentagem.

Gráfico 4 - Desempenho das ferramentas de firewall



	RAM em execução (utilização em MB)	Armazenamento (utilização em MB)	CPU (utilização em %)	Bateria (utilização em %)
Droidwall	6,26	1,06	0,4	0,03
Root Firewall	13,49	1,11	0,65	0,09
Bluetooth Firewall Trial	10,25	0,745	11,12	0
Avast	11,53	5,7	2,73	0,17

Fonte: da Autora

Quanto ao critério desempenho das ferramentas de firewall, podemos averiguar que na utilização de memória RAM do dispositivo, a ferramenta a mais utilizar é o Root Firewall, e a que menos utiliza a memória é a Droidwall. Em utilização da memória para armazenamento, a ferramenta a mais utilizar foi o Avast com 5,7 MB e a que menos utiliza é a Bluetooth Firewall Trial com 0,745KB. Quanto ao uso da CPU, a mais utilizar é a ferramenta Bluetooth Firewall Trial com 11,12% e que menos utiliza é o Droidwall com 0,4%. Na questão de uso da bateria a que menos consome é o Bluetooth Firewall Trial com 0% e a que mais consome é a Avast com 0,17%, porém vale ressaltar que a ferramenta Avast, além de utilizar antivírus também possui integrado o serviço de firewall.

9.2.3 Avaliação das melhores ferramentas de *firewall*

Pode-se chegar à conclusão, observando os resultados obtidos, que a melhor ferramenta de firewall é o Droidwall, por obter um bom resultado de eficiência, utilizar menos recursos do dispositivo, além de possuir suporte para a criação de regras personalizáveis.

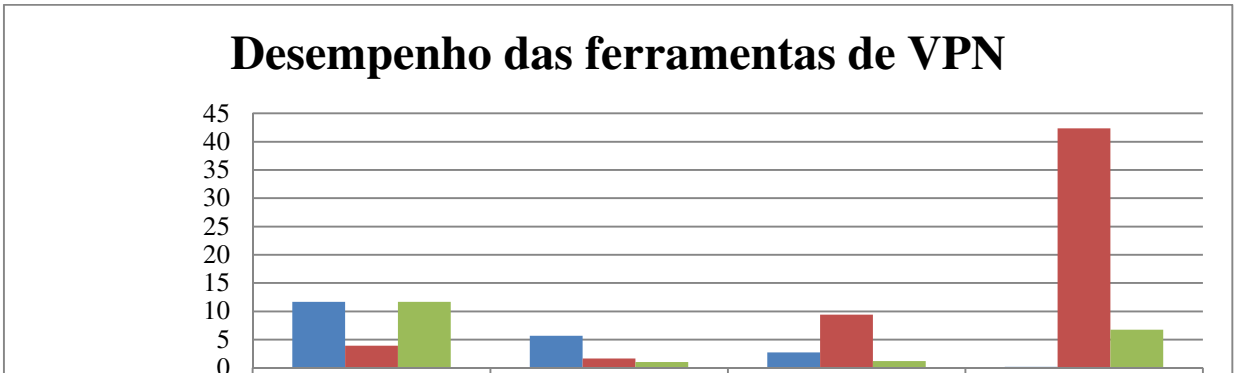
9.3 COMPARAÇÃO ENTRE AS FERRAMENTAS DE VPN

A seguir será demonstrada através de gráficos uma comparação dos critérios usados nos testes de avaliação das ferramentas de segurança de VPN, sendo o critério comparado o de desempenho.

9.3.1 Comparativo das ferramentas de VPN no critério desempenho

Na tabela abaixo são apresentados os resultados quanto ao desempenho das ferramentas de VPN, a utilização de memória RAM da ferramenta em execução definida em MB, a utilização de armazenamento da ferramenta no dispositivo em MB e a utilização da CPU e da bateria em porcentagem.

Gráfico 5 - Desempenho das ferramentas de VPN



	RAM em execução (utilização em MB)	Armazenamento (utilização em MB)	CPU (utilização em %)	Bateria (utilização em %)
VPN One Click	11,65	5,7	2,73	0,17
Hiden VPN	3,91	1,67	9,41	42,35
Viatun 4 VPN	11,65	1,03	1,22	6,73

Fonte: da Autora

Quanto ao critério desempenho das ferramentas de VPN, pode-se averiguar que na utilização de memória RAM do dispositivo, as ferramentas a mais utilizar são o VPN One Click e Viatun 4VPN, e a que menos utiliza memória é a *Hiden* VPN. Em utilização da memória para armazenamento, a ferramenta a mais utilizar foi o VPN *One Click* com 5,7 MB e a que menos utiliza é a ferramenta Viatun 4 VPN com 1,03MB. Quanto ao uso da CPU, a que mais consome processamento é a ferramenta *Hiden* VPN com 9,41% e a que menos utiliza é o Viatun 4 VPN com 1,22%. Na questão de uso da bateria a que menos consome é o VPN *One Click* com 0,17% e a que mais consome é a ferramenta Hiden VPN com 42,35%.

9.3.2 Avaliação da melhor ferramenta de VPN

Na avaliação das melhores ferramentas de VPN, a Viatun 4 VPN, ganhou no quesito de desempenho, pois esta foi a que menos utilizou recursos do dispositivo, já a ferramenta que possui mais recursos do que as outras ferramentas foi a VPN *One Click*, pois esta possui conexão com servidores remotos de outros países, tipos de planos diferenciados, além de possuir opção de escolher o protocolo para realizar a conexão com servidores remotos e obter um túnel criptografado.

9.4 AVALIAÇÃO TOTAL DAS FERRAMENTAS

Com as ferramentas testadas, podemos chegar à conclusão de que com a utilização de um conjunto de ferramentas o dispositivo poderá estar seguro de ataques aqui demonstrados, pois com a combinação de um antivírus robusto, de um firewall com bloqueio para rede *wi-fi* e 2g/3g, de um firewall para bloqueio de *bluetooth* e de uma rede VPN o dispositivo ficará mais protegido. Porém é necessário levar em consideração que a utilização de todas as ferramentas poderá acarretar uma perda de desempenho do dispositivo, também há o fato de que para alguns serviços é necessário a liberação do superusuário *root* e em alguns casos a utilização de *ROMs* customizadas.

10 CONSIDERAÇÕES FINAIS

Com o presente trabalho foi possível identificar quais as vulnerabilidades e verificar as opções existentes quanto à segurança da Plataforma *Android*. Cabe ressaltar que com este trabalho não foi possível averiguar a totalidade destes elementos, já que a identificação de vulnerabilidades é processo constante, pois a utilização de dispositivos móveis está crescendo e as opções de segurança para a plataforma *Android* são muitas, sendo muito difícil a averiguação de todas estas ferramentas em um curto prazo.

Quanto aos testes realizados com as ferramentas de segurança, pode-se verificar que estas são diferenciadas das ferramentas de utilização costumeira em computadores comuns, uma vez que estas são desenvolvidas com a intenção de suprir as necessidades do dispositivo e da plataforma. Nas ferramentas de antivírus, cada aplicativo possui um pacote de serviços diferente de outro, algumas das ferramentas testadas em sua versão gratuita, possuem mais serviços do que outra. As ferramentas de *firewall* possuem um funcionamento exclusivo para o dispositivo móvel, sem muitas opções de bloqueio, como, por exemplo, o de bloquear aplicativos ou serviços somente para a rede *Wifi* ou 2G/3G, realizando bloqueios apenas selecionando as opções já existentes, onde cada serviço e aplicativo são listados. Quanto a estas ferramentas, as únicas a possuírem a criação de regras customizadas, porém ainda limitadas, são as ferramentas DroidWall, que tem na criação de regra em forma de *script*, sendo necessário um conhecimento maior por parte do usuário, e a ferramenta Avast que possui integrado em seu antivírus o serviço de *firewall*. Um ponto negativo quanto ao serviço de *firewall* é a de que ele não funciona sem a liberação do superusuário *root* e também em algumas versões de *ROMs* criadas pelos fabricantes não é possível a utilização do firewall, sendo necessário ainda utilizar uma *ROM* customizada, sendo que estes processos só são realizados por usuários avançados. Quanto ao serviço de VPN, é possível a conexão em redes VPN remotas, podendo até utilizar uma rede de outro país, porém as conexões não foram estabelecidas em nenhum serviço de VPN das ferramentas testadas.

Dos testes de invasão realizados, foi possível averiguar que estando em uma rede, através de aplicativo, por meio de um site ou via *bluetooth*, é possível obter acesso aos dados do dispositivo. Os testes realizados na versão 2.3.6 do *Android* demonstram que grande parte da população que utiliza dispositivos com esta versão, ou anterior,

estão vulneráveis a ataques, sendo que dentre as ferramentas testadas, muitas não conseguiram encontrar ou bloquear os ataques realizados. A maioria das ferramentas desenvolvidas para a segurança dos dispositivos, conforme avaliação realizada ao longo do presente trabalho, não são plenamente eficazes, podendo ainda detectar algumas vulnerabilidades, porém não em sua totalidade. Entrentanto, com a utilização de uma combinação de *firewall*, antivírus e VPN, o dispositivo estará mais seguro quanto a ataques, pois uma ferramenta complementa a outra.

Com base nos resultados encontrados no presente estudo, sugere-se que em trabalhos futuros nesta mesma área, podem ser realizados estudos no sentido de:

- Criação de ferramentas gratuitas mais eficazes quanto à segurança do dispositivo, uma vez que as ferramentas gratuitas testadas não foram completamente eficazes.
- Criação de ferramentas para a realização de testes de invasão em dispositivos móveis.
- Comparação de mais ferramentas de segurança envolvendo versões pagas e gratuitas.

Com o trabalho realizado foi possível averiguar que a plataforma *Android* possui algumas deficiências quanto à segurança dos dispositivos, principalmente aqueles que não utilizam nenhum tipo de ferramenta de segurança. Portanto, é necessário uma maior atenção de usuários e profissionais da tecnologia quanto ás vulnerabilidades existentes na plataforma *Android,* bem como das ferramentas disponíveis para sua segurança.

REFERÊNCIAS

ANDROID bugs allow malicious app to install on your device without your permission. Disponível em: < http://phandroid.com/2011/09/20/2-android-bugs-allow-malicious-apps-to-install-on-your-device-without-your-permission/ > Acesso em: 22 abr. 2012.

ANDROID Security Overview. Disponível em: <http://source.android.com/tech/security/index.html#android-platform-security-architecture>. Acesso em: 30 mai. 2012.

ATAQUES de hacker a smartphones devem aumentar muito em 2012, dizem especialistas do IEEE. Tecmundo, 2012. Disponível em: <http://www.tecmundo.com.br/celular/17334-ataques-hacker-a-smartphones-devem-aumentar-muito-em-2012-dizem-especialistas-do-ieee.htm>. Acesso em: 22 abr. 2012.

AVTEST. Test report: Anti-Malware solutions for Android. Alemanha, 2012. Disponível em:< http://www.av-test.org/fileadmin/pdf/avtest_2012-02_android_anti-malware_report_english.pdf >. Acesso em: 21 abr. 2012.

CARVALHO, Luciano Gonçalves. *Segurança de Redes*. Rio de Janeiro: Editora Ciência Moderna Ltda., 2005.

CHESWICK, William R. *Firewalls e segurança na Internet*: repelindo o hacker ardiliso. Steven M. Belovin e Aviel D. Rubin. Trad. Edson Furmankiewiez – 2 ed. Porto Alegre : Bookman, 2005.

CIBRÃO, Daniel; GONÇALVES, Rui. Segurança no Android. Disponível em: <http://web.fe.up.pt/~jmcruz/ssi/trabs-als/final/G4T10-android-final.pdf>. Acesso em: 16 abr. 2012.

CRIPTOGRAFIA. Disponível em: <http://www.infowester.com/criptografia.php>. Acesso em: 03 de abr. 2012.

DEVELOPER Android. Disponível em: < http://developer.android.com/reference/ javax/crypto/package-summary.html >. Acesso em: 03 de abr. 2012.

DEVICE Administration. Disponível em: <https://developer.android.com/guide/topics/admin/device-admin.html >. Acesso em: 30 mai. 2012.

ESET. Trends for 2013: Astounding growth of mobile malware. Disponível em: <http://go.eset.com/us/resources/white-papers/Trends_for_2013_preview.pdf>. Acesso em: 13 jul. 2013.

ESTUDO da IDC revela que foram vendidos aproximadamente 9 milhões de smartphones no Brasil em 2011. Disponível em: <http://www.idclatin.com/news.asp?ctr=bra&year=2012&id_release=2213>. Acesso em 04 abr. 2012.

FUNCTION setcookies. Dispoível em: < http://php.net/manual/pt_BR/function.setcookie.php >. Acesso em: 04 abr. 2013.

GIAVAROTO, Sílvio César Roxo. SANTOS, Gerson Raimundo dos. Backtrack Linux – Auditoria e Teste de Invasão em Redes de Computadores. Rio de Janeiro: Editora Ciência Moderna Ltda., 2013.

JAVVIN Technologies Inc.Network Dictionary. Disponível em: <http://books.google.com.br/books?id=On_Hh23IXDUC&printsec=frontcover&dq=network+dictionary&hl=pt-BR&sa=X&ei=V43nUf6uBIa-9gSV64HoAw&ved=0CDUQ6AEwAA>. Acesso em 17 jul. 2013.

KASPERSKY. Relatório virologia móvel 2011 e tendências 2012. Disponível em: <http://www.kaspersky.com/pt/news?id=207576091>. Acesso em: 15 mai. 2012.

LECHETA, Ricardo R. *Google Android* : aprenda a criar aplicações para dispositivos móveis com o Android SDK . 2. ed. São Paulo: Novatec Editora, 2010.

MCAFEE. Mobilidade e segurança. Disponível em: <http://www.mcafee.com/br/resources/reports/rp-cylab-mobile-security.pdf>. Acesso em: 15 mai. 2012.

MCAFEE. Proteção de dispositivos móveis: presente e futuro. Disponível em: <http://www.mcafee.com/br/resources/reports/rp-securing-mobile-devices.pdf>. Acesso em: 15 mai. 2012.

MORAES, Paulo. *Mente anti-hacker: proteja-se*. Rio de Janeiro : Brasport, 2011.

MORAZ, Eduardo. *Treinamento profissional anti-hacker*. São Paulo: Digerati Books, 2006.128p.

MORIMOTO, Carlos Eduardo. *Redes, guia prático*. Porto Alegre : Sul Editores, 2008.

MYERSON, Judith M. Crie políticas de segurança para dispositivos móveis. 2011. Disponível em: <http://www.ibm.com/developerworks/br/cloud/library/cl-mobilesecuritypolicy/index.html>. Acesso em: 21 abr. 2012.

PLAY. Disponível em:< https://play.google.com >. Acesso em: 10 jun. 2013.

PEREIRA, Lúcio Camilo Oliva; SILVA, Michel Lourenço da Silva. *Android para Desenvolvedores*. Rio de Janeiro: Brasport, 2009.

ROTEAMENTO de pacotes e nat no Linux. Disponível em:< http://www.vivaolinux.com.br/dica/Roteamento-de-pacotes-e-NAT-no-Linux > Acesso em: 11 jul. 2013.

RODRIGUES, Guilherme Rodrigues e. Smartphones e suas tecnologias. Disponível em: <http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=books&cd=1&ved=0CEEQFjAA&url=http%3A%2F%2Fwww.tcc.sc.usp.br%2Ftce%2Fdisponiveis%2F18%2F180450%2Ftce-23042010-094332%2Fpublico%2FRodrigues_Guilherme_Rodrigues_e.pdf&ei=5U_iT9uoD4Ku9AT72ISHCA&usg=AFQjCNHj1zzW8ZAYmUfZ4QYtyaocXIduPQ&sig2=yDgwpP2rnoXt5Sv0iXREfg>. Acesso em: 10 mai. 2012.

SACHSE, Nelson Ricardo Santos. Avaliação Comparativa do Modelo de Segurança do Android. 2010. Disponível em:

<http://bdigital.ufp.pt/bitstream/10284/1960/2/DM_12464.pdf>. Acesso em: 04 abr. 2012.

SCARFONE, Karen; SOUPPAYA, Murugiah; CODY, Amanda; OREBAUGH, Angela. Technical Guide to Information Security Testing and Assessment – Recommendations of the National Institute of Standards and Technology. Gaithersburg: NIST, 2008.

SECURITY and Permissions. Disponível em: <http://developer.android.com/guide/topics/security/security.html>. Acesso em: 30 mai. 2012.

SILVA, Gilson Marques da. *Segurança em Sistemas Linux*. Rio de Janeiro: Editora Ciência Moderna Ltda., 2008.

SMS e MMS. Disponível em:< http://seumicroseguro.com/?s=SMS+e+MMS >. Acesso em: 30 mai. 2012.

SOURCE Android. Disponível em:< http://source.android.com/source/index.html>. Acesso em: 30 mai.2012

STALLING, William. Criptografia e segurança de redes / William Stallings; tradução Daniel Vieira; revisão técnica Ákio Barbosa e Marcelo Succi. – 4. Ed. – São Paulo : Pearson Prentice Hall, 2008.

VULNERABILITY list. Disponível em: < http://www.cvedetails.com/vulnerability-list/vendor_id-1224/product_id-19997/Google-Android.html >. Acesso em: 30 mai. 2012.

ANEXOS

ANEXO A – IMAGEM DO TESTE DA EMPRESA AVTESTE CORRESPOSNDENTE AOS ANTIVÍRUS DO PLATAFORMA *ANDROID*

	Product	Average Family Detection	
A	avast! Free Mobile Security		>90%
A	Dr.Web anti-virus Light		
A	F-Secure Mobile Security		
A	IKARUS mobile.security LITE		
A	Kaspersky Mobile Security		
A	Lookout Security & Antivirus		
B	McAfee Mobile Security		
B	MYAndroid Protection		
B	NQ Mobile Security		
A	Zoner AntiVirus Free		
A	AegisLab Antivirus Free		>65%
A	AVG Mobilation Anti-Virus Free		
A	Bitdefender Mobile Security		
B	BullGuard Mobile Security		
B	Comodo Mobile Security		
A	ESET Mobile Security		
A	Norton Mobile Security Lite		
A	Quick Heal Mobile Security		
A	Super Security		
B	Total Defense Mobile Security		
A	Trend Micro Mobile Security		
A	Vipre Mobile Security (BETA)		
A	Webroot SecureAnywhere		
B	BluePoint Security Free		>40%
B	G Data Mobilesecurity		
B	Kinetoo Malware Scan		
B	ALYac Android		>0%
B	Android Antivirus		
B	Android Defender Virus Shield		
B	Antivirus Free		
B	BlackBelt AntiVirus		
B	CMC Mobile Security		
B	Fastscan Anti-Virus Free		
B	GuardX Antivirus		
B	MobiShield Mobile Security		
B	MT Antivirus		
B	Privateer LITE		
B	Snap Secure		
B	TrustGo Mobile Security		
B	LabMSF Antivirus beta		0
B	MobileBot Antivirus		