Universidade de Caxias do Sul

Vinícius Lahm Perini

Integração de Ferramentas de Administração e Segurança BYOD

Caxias do Sul

2017

Vinícius Lahm Perini

Integração de Ferramentas de Administração e Segurança BYOD

Trabalho de Conclusão de Curso de Ciência da Computação apresentado ao Centro de Ciências Exatas e da Tecnologia da Universidade de Caxias do Sul, como requisito para a disciplina Projeto de Diplomação II.

Orientadora: Profa. Dr. Maria de Fátima Webber do Prado Lima

Caxias do Sul

2017

RESUMO

O número de dispositivos móveis de uso pessoal conectados na internet está crescendo constantemente e sua utilização no âmbito profissional aumentou. O BYOD tem ganhado popularidade justamente porque aumenta a satisfação do empregado e diminui os custos de infraestrutura para o empregador. Uma vez que a corporação tenha adotado tal modelo de negócio, é preciso atentar ao fato de que seus dados sensíveis podem estar sendo acessados através de dispositivos que não estão de acordo com as políticas de segurança previamente definidas. O maior desafio para as corporações que querem se modernizar e adotar este novo conceito é como manter os seus dados seguros. Apesar dos benefícios obtidos com a utilização do conceito BYOD, a segurança da informação precisa ser remodelada para que garanta o não vazamento de informações sensíveis. Com o intuito de promover um protótipo de *software* que tenha capacidade de oferecer uma boa solução para a administração e segurança de redes BYOD, foram estudadas algumas ferramentas disponíveis no mercado. Dentre diversas ferramentas citadas neste trabalho, algumas foram selecionadas para compor o *software* de integração de ferramentas, tais como *NetworkMiner, Universal Password Manager, Password Strength Meter, NetCalculator, jNetMap e IP Monitor*. Este *software* tem como uma de suas características principais a facilidade de utilização, além de seu planejamento, visando facilitar a compatibilidade com diversas plataformas e sistemas operacionais.

Palavras-chave: BYOD. Integração de Ferramentas *Open-Source*. Administração e Segurança de Redes. Dispositivos Móveis.

ABSTRACT

The number of mobile devices for personal use connected on the internet is constantly growing and usage in the professional scope has increased. BYOD has gained popularity precisely because it increases employee satisfaction and lowers infrastructure costs for the employer. Once the corporation has adopted such a business model, you have to be aware that your sensitive data may be accessed through devices that do not conform to the security policies previously defined. The biggest challenge for corporations who want to modernize and adopt this new concept is how to keep their data secure. Despite the benefits of using the BYOD concept, information security needs to be remodeled to ensure that no sensitive information is leaked. In order to promote a prototype software that has the capacity to offer a good solution for the administration and security of BYOD networks, some tools available in the market have been studied. Among several tools cited in this paper, some were selected to compose the tool integration software such as NetworkMiner, Universal Password Manager, Password Strength Meter, NetCalculator, jNetMap and IP Monitor. This software has as one of its main features the ease of use, in addition to its planning to facilitate compatibility with various platforms and operating systems.

Keywords: BYOD. Open-Source Tools Integration. Administration and Network Security. Mobile devices.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

Sigla	Significado
.NET	*DotNET*
AES	*Advanced Encryption Standard*
AES-SIV	*Advanced Encryption Standard - Synthetic Initialization Vector*
AESCTR	*Advanced Encryption Standard – Counter Mode*
API	*Application Programming Interface*
ARP	*Address Resolution Protocol*
BYOD	*Bring Your Own Device*
BYOT	*Bring Your Own Technology*
CSPRNG	*Cryptographically Secure Pseudo-Random Number Generator*
CYOD	*Choose Your Own Device*
DDOS	*Distributed Denial-Of-Service*
DECNET	Conjunto de protocolos de redes criado pela Digital Equipment Corporation (DEC).
DLC	*Data Link Control*
DLL	*Dynamic-Link Library*
DLP	*Data Link Protocol*
DNS	*Domain Name System*
DRP	*Disaster Recovery Plan*
EMM	*Enterprise Mobility Management*
EXE	Extensão de arquivo que denota um arquivo executável no ambiente Windows.
GUI	*Graphical User Interface*
HTTP	*Hypertext Transfer Protocol*
HYOD	*Here is Your Own Device*
ICMP	*Internet Control Message Protocol*
IDE	*Integrated Development Environment*
IP	*Internet Protocol*
IPX	*Internetwork Packet Exchange*
JVM	*Java Virtual Machine*
KEK	*Key Encryption Key*

MAC	*Media Access Control*
MDM	*Mobile Device Management*
MVC	*Model View Controller*
NIDS	*Network Intrusion Detection System*
OYOD	*Own Your Own Device*
SLA	*Service-Level Agreements*
SSH	*Secure Shell*
STAT	*Schedule Test and Assessment Tool*
TCP	*Transmission Control Protocol*
TI	Tecnologia da Informação
TVR	Testes de Vulnerabilidade de Rede
UPD	*User Datagram Protocol*
VPN	*Virtual Private Network*
WAP	*Wi-Fi Protected Access*
WEP	*Wired Equivalent Privacy*

SUMÁRIO

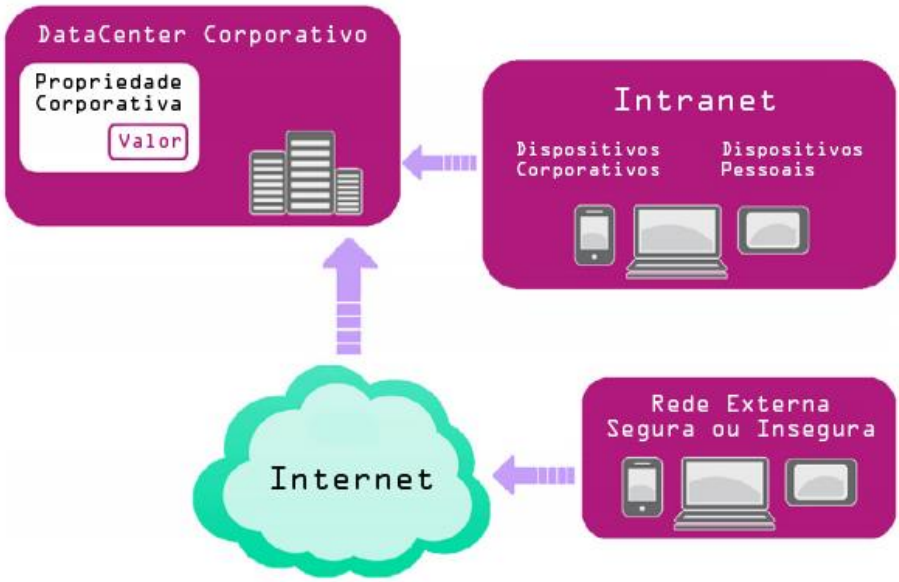
1. INTRODUÇÃO

O constante desenvolvimento de novas tecnologias de informação está fazendo com que o modo como o usuário utiliza seus dispositivos mude drasticamente. Se antes computadores e celulares eram artigos de luxo para usuários domésticos, hoje a realidade é outra. O número de dispositivos por usuário aumentou e um dos principais motivos para este progressivo crescimento é chamado Internet das Coisas.

Internet das Coisas é o termo adotado para classificar a comunicação e interação entre os mais diversos aparelhos utilizados no dia a dia, sejam eles domésticos ou empresariais. Essa conexão entre dispositivos (IoT) é definida como a atual revolução pelo visível crescimento de dispositivos habilitados para a Internet que podem se comunicar uns com os outros e com outros *gadgets* também habilitados para a Web. Internet das Coisas refere-se a um estado no qual Coisas, por exemplo, objetos, ambientes, veículos e roupas terão cada vez mais informações associadas a si, com a possibilidade de sentir, comunicarem-se, e produzir novas informações, tornando-se parte integrante da Internet, conforme afirmação de *Technology Strategy Board* (2013).

Com este conceito, usuários tem mais flexibilidade, pois agora podem usar seus próprios dispositivos para acessar recursos empresariais e continuar suas atividades fora do local de trabalho (ELSEVIER, 2015). As empresas passaram a adotar novas medidas para se modernizarem, como por exemplo, o BYOD (*Bring Your Own Device)*, ou seja, Traga Seu Próprio Dispositivo (Figura 1). Este é o conceito de usar o celular, *tablet*, *notebook* e outros aparelhos móveis de uso pessoal, em vez de utilizar a infraestrutura de TI da empresa.

Figura 1 - Topologia de rede corporativa com BYOD.



Fonte: ELSEVIER, 2015.

Diante disso, manter a privacidade do usuário e a segurança dos dados corporativos são os dois aspectos principais quando o assunto é segurança de TI. No modelo BYOD, essa questão se torna ainda mais relevante, pois o administrador da rede precisará manter a integridade de uma rede com conexões de diversos sistemas operacionais e diversas possíveis ameaças.

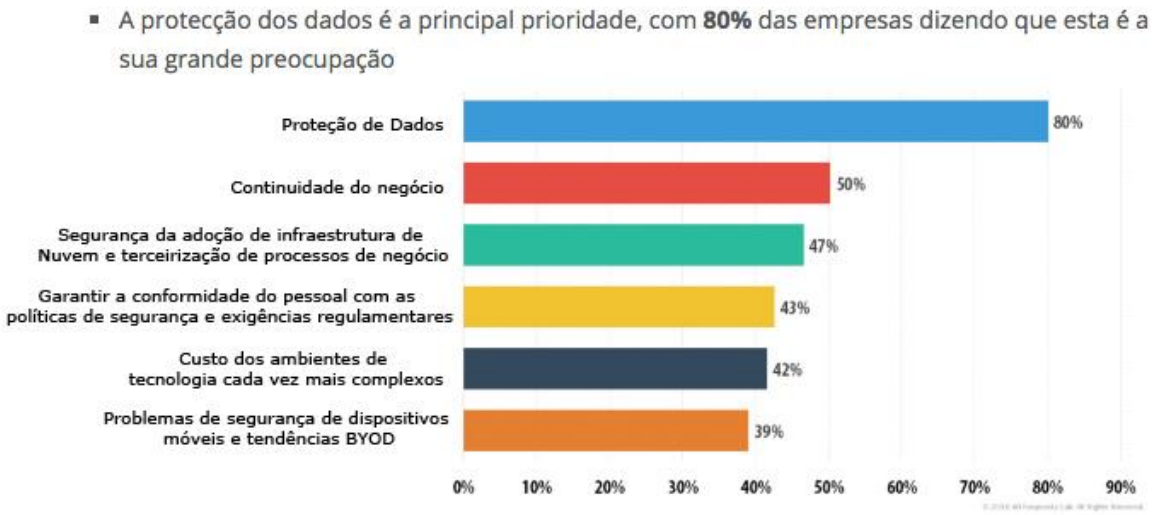
Em uma entrevista feita com profissionais de TI de diversos países, foi constatado que 65% desses consideram que existem riscos decorrentes de sua incapacidade de controlar os dispositivos e aplicativos dos funcionários (COMPUTER WORLD, 2017). Essa é a principal porta de entrada para *hackers*, pois o funcionário pode estar usando um equipamento próprio que está contaminado e acabar oferecendo um perigo para todos os outros usuários conectados.

Em novembro de 2016 ocorreu um ataque de negação de serviço (DDoS) à Oracle Dyn, empresa que oferece serviços de DNS, que consequentemente deixou indisponível por algumas horas quase uma centena de serviços de entretenimento, sites de compras ou de notícias, e até algumas agências governamentais. Numa investigação, identificou-se um *malware* que havia contaminado em série o sistema operacional de diversos dispositivos como brinquedos e fornos de micro-ondas (CEGALI, 2017). Um mês depois, pesquisadores da *SEC Consult* revelaram que 80 modelos de câmeras de segurança profissionais da Sony,

usadas principalmente por corporações, tinham contas de *backdoor* que permitiam o acesso à rede local das mesmas.

Segundo a recente pesquisa feita pela *Kaspersky Lab*, o compartilhamento inadequado de dados via dispositivos móveis representa a ameaça mais difícil de administrar, com 54% dos votos. Analisando as prioridades das corporações, é notável a proteção dos dados como a principal preocupação, porém é preciso alertar ao fato de que a segurança de dispositivos BYOD está em último lugar na pesquisa, como ilustrado na Figura 2.

Figura 2 - Dados da pesquisa da Kaspersky Lab



Fonte: CIO, 2016.

É cada vez mais comum dispositivos se conectarem à internet e por isso as empresas devem atentar nas novas formas de ameaças. Com locais de trabalho mais modernos, tecnologias como realidade virtual, dispositivos de Internet das Coisas, *smartphones* e vestíveis estão conectados na rede executando aplicativos e soluções na nuvem. Isso representa uma mudança na forma como a segurança de TI deve ser planejada dentro de uma corporação, trocando o foco que antes era nos dispositivos *endpoint* para a proteção do usuário e sua privacidade. O investimento em segurança focada no modelo BYOD ajuda a manter seguros os dados corporativos e, por consequência, os dados pessoais do usuário, que será alertado sobre a ameaça em seu dispositivo.

Existem variadas ferramentas de administração e de segurança focadas neste modelo de rede. Algumas são pagas, outras são soluções *open-source*, mas todas podem ser muito

eficientes no combate de novas ameaças quando usadas por corporações. O auxílio e as funcionalidades destas ferramentas serão apresentados neste projeto.

1.1 PROBLEMA E QUESTÃO DE PESQUISA

A cada ano que passa mais empresas adotam o modelo BYOD de trabalho. Algumas delas motivadas pela economia de não precisar investir em equipamentos, outras apenas abrindo uma nova escolha ao trabalhador. Consequentemente, esse processo exige novas demandas na infraestrutura e na administração da rede.

As aplicações responsáveis pela segurança da rede que eram utilizadas anteriormente precisam de atualizações. As ferramentas disponíveis para a segurança BYOD realizam tarefas específicas em diferentes sistemas operacionais e em diferentes tipos de dispositivos. Esta situação acaba causando problemas ao profissional de TI, que precisa dominar diferentes *softwares* de diversos fornecedores.

Desta forma, a principal motivação deste trabalho é a pesquisa e o desenvolvimento de um protótipo com capacidade de fazer com que diversas ferramentas de código aberto interoperem, possibilitando uma integração entre elas, facilitando a usabilidade e melhorando a experiência do profissional de TI responsável pela rede e pela segurança de dados.

Com a definição do problema citado acima, a questão de pesquisa pode ser definida por: Qual boa prática pode-se utilizar para implementar um *software* que consiga entregar uma interface simples e otimizada, que forneça os recursos administrativos e de segurança para redes BYOD de maneira integrada e que suporte aplicações *open-source* com diversas linguagens de programação, e assim, facilite as tarefas diárias do profissional de TI.

1.2 OBJETIVO GERAL

O objetivo geral deste trabalho é propor um protótipo de integração de *sotfwares open-source* voltado para a segurança e a administração de redes corporativas que siga a filosofia BYOD. Durante o processo serão analisadas as soluções disponíveis no mercado e após, através do protótipo proposto, será possível estabelecer uma boa usabilidade e consequentemente maior praticidade para o profissional de TI. Com o intuito de atingir o objetivo geral, alguns passos menores serão definidos como os objetivos específicos. São eles:

- Realizar um estudo sobre o modelo BYOD e seus principais problemas de segurança.
- Realizar uma pesquisa de ferramentas *open-source* para administração e segurança de redes corporativas.
- Modelagem dos principais recursos que serão necessários para ampliar a abrangência dos *softwares*.
- Desenvolvimento de uma aplicação capaz de integrar tais ferramentas baseado na modelagem de recursos.
- Testar e analisar os resultados obtidos pela aplicação.

1.3 METODOLOGIA

A metodologia adotada neste trabalho é composta por quatro etapas. Na primeira etapa será feito um estudo teórico sobre as redes BYOD focado em administração e segurança. Além disso, uma busca por ferramentas de proteção e gerenciamento será realizada.

Na segunda etapa do processo, será feito um breve estudo sobre as ferramentas existentes. Deste modo será possível identificar quais destas ferramentas previamente selecionadas serão escolhidas. O *software* de integração será uma aplicação *Desktop* capaz de se comunicar, através de *frameworks*, com as outras linguagens de programação que possam ser encontradas nas ferramentas *open-source* escolhidas.

Depois de coletar as informações acima e analisar as mesmas, será o momento de desenvolver o protótipo de camada de interoperabilidade. Nesta etapa todos os conceitos e tecnologias necessários serão aplicados durante o processo de programação.

Finalmente, na quarta etapa será feito um teste, no Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, verificando se o *software* cumpre os seus requisitos e se realmente propõe uma facilidade durante o trabalho do profissional de TI responsável pela administração e segurança dos dados corporativos.

1.4 ESTRUTURA DO TRABALHO

O trabalho está estruturado da seguinte maneira: Capítulo 2 (BYOD - *Bring Your Own Device*) no qual o conceito é apresentado. Suas vantagens e desvantagens são listadas e,

além disso, os seus principais desafios na questão de segurança da informação corporativa e pessoal. No Capítulo 3 (Caracterização da Pesquisa) é definida qual a metodologia de pesquisa adotada durante o desenvolvimento do projeto e uma classificação é sugerida para os *softwares* BYOD presentes no mercado. Diversas opções de ferramentas disponíveis são descritas neste capítulo. Finalmente, as ferramentas que foram escolhidas para o trabalho são citadas. No Capítulo 4 (Estruturação do Protótipo), é apresentada a proposta de integração de ferramentas de administração e segurança BYOD, tal como sua arquitetura de *software* e modelagem. No Capítulo 5 (Desenvolvimento do Protótipo) é apresentada a estrutura de desenvolvimento utilizada, como a integração foi estabelecida e quais classes foram criadas. O Capítulo 6 (Testes e Resultados) demonstra a quais testes a ferramenta desenvolvida foi submetida e seus respectivos resultados. No Capítulo 7 (Conclusão) são apresentadas as conclusões gerais do estudo e quais melhorias podem ser desenvolvidas além de trabalhos futuros.

2. BYOD (Bring Your Own Device)

Bring Your Own Device é a preferência pela utilização de dispositivos pessoais para exercer tarefas profissionais. Desta forma, o funcionário acessa os dados corporativos, através do seu *tablet*, por exemplo, para realizar suas demandas diárias. Segundo pesquisa realizada por *Harris Interactive* e ESET, mais de 80% dos empregados utilizam algum tipo de dispositivo eletrônico pessoal para tarefas profissionais (MORROW, 2012). Com isso, o funcionário tem maior liberdade e pode utilizar a marca e/ou a tecnologia que mais lhe convém. Desta maneira, um ambiente de trabalho mais conveniente é oportunizado. Este modelo de rede corporativa proporciona redução nos investimentos da empresa em *hardware*, e amplia a produtividade (THOMSON, 2012). Porém, a principal característica do modelo de rede BYOD é somente permitir a conexão de um dispositivo se ele estiver de acordo com as políticas de segurança pré-definidas pelos administradores.

A Tabela 1 mostra um comparativo entre as vantagens e as desvantagens do modelo BYOD.

Tabela 1- Comparativo BYOD.

Vantagens	**Desvantagens**
Diminui custos, principalmente em pequenas empresas.	A segurança dos dados é um desafio para o setor de TI.
Possibilita ao empregado escolher sua tecnologia e consequentemente melhora a sua satisfação no trabalho.	Processo mais rigoroso para manter a confidencialidade de informações corporativas.
Organização se beneficia com o uso de tecnologias atuais e novas possibilidades que aumentam a produtividade.	Departamento de TI precisa manter uma lista dos dispositivos pessoais que estão habilitados para a rede BYOD. Essa lista deve ser constantemente monitorada e atualizada.
Funcionário tem maior habilidade com o dispositivo, facilitando a solução autônoma de problemas.	Alta variedade de dispositivos podem gerar problemas de compatibilidade e de desempenho.

Com base em: CISCO, 2012 e CIO, 2012.

Apesar do corte de custo em *hardware*, a corporação terá uma maior demanda no setor de infraestrutura de TI, com profissionais capazes de fornecer suporte a diferentes plataformas. Além disso, a empresa precisa aumentar sua atenção na questão de segurança de informação. O BYOD pode facilitar o vazamento de dados, uma vez que os dispositivos não são diretamente gerenciados pelos administradores da rede.

O uso de sistemas de TI e soluções de TI por parte do funcionário, sem aprovação organizacional explícita, caracterizam a *Shadow IT*[1] (SOMMERFELD, 2015). Por exemplo, um funcionário pode utilizar, através do próprio dispositivo, serviços em nuvem durante o trabalho. Estes serviços podem ter as mais variadas finalidades, como por exemplo, gerenciamento de agenda, projetos, tarefas, entre outros. Porém, os dados utilizados nestas ferramentas talvez não estejam protegidos apropriadamente ou em conformidade aos padrões de segurança da organização. Isso normalmente ocorre quando a empresa não fornece todas as soluções necessárias para o empregado, que acaba buscando aplicações de terceiros.

2.1 TECNOLOGIAS ENVOLVIDAS

O modelo BYOD abriu caminho para outras tecnologias serem adotadas no setor corporativo. Diversas ferramentas estão diretamente envolvidas com esta forma de trabalho e resultaram em novos conceitos como o BYOT (*Bring Yor Own Technology*) e o BYOS (*Bring Your Own Software*). Estes dois modelos sugerem que o funcionário utilize uma tecnologia ou *software*, respectivamente, que a empresa não conheça ou não possua para realizar suas obrigações. Além destes modelos, é possível destacar várias outras tecnologias utilizadas juntamente com o BYOD. São elas: Computação em Nuvem, Virtualização de Aplicações, Virtualização de *Desktop*, entre outras.

A computação em nuvem pode ser entendida como um ambiente de computação composto por inúmeros servidores, virtualizados ou físicos, ou um conjunto de soluções de TI com capacidade de processamento, armazenamento, aplicações, plataformas e serviços disponibilizados através da internet (TAURION, 2009). Ela é responsável por disponibilizar o acesso aos dados corporativos via dispositivos pessoais dos empregados quando eles estão fora do ambiente de trabalho.

[1] *Shadow IT* é o termo utilizado para definir o uso indevido de *softwares* pelo funcionário. Mais informações em: *The Hidden Truth Behind Shadow IT - Six trends impacting your security posture*, McAfee, novembro de 2013.

A virtualização de aplicações é utilizada para facilitar a criação de um escritório remoto, por exemplo. Se a aplicação não interagir de forma significativa com o *kernel* do sistema, ela pode ser virtualizada gerando flexibilidade e agilidade. A alocação de recursos de TI dentro da empresa pode ser feita de maneira mais prática (CIO, 2011). Esta tecnologia consome os recursos de processamento do dispositivo de destino, como por exemplo, um *notebook* pessoal.

A virtualização de *desktop* é a tecnologia capaz de emular uma máquina na rede utilizando os servidores que a própria corporação possui. Desta forma o funcionário tem a possibilidade de acessar os dados remotamente através do seu dispositivo preferido. Esta modalidade permite um gerenciamento mais fácil por parte dos administradores. As máquinas virtuais são criadas a partir de modelos pré-definidos com *softwares* atualizados e configurações de rede prontas.

Para que o funcionário realize a conexão de qualquer local, isto é, de fora da corporação, alguns protocolos de comunicação são utilizados. Existem duas soluções comumente empregadas, que são: as já tradicionais VPN (*Virtual Private Network*) e as novas MDM (*Mobile Device Management*). A VPN possibilita que duas ou mais máquinas se comuniquem através de uma rede pública, neste caso a Internet, criando um túnel entre elas e criptografando seus dados. Já as soluções MDM são resultados de um aglomerado de tecnologias que buscam abraçar o mercado móvel. Estas soluções são focadas em promover o gerenciamento do dispositivo, o gerenciamento de usuário e a conexão móvel (DELANEY, 2012). As soluções MDM trabalham nativamente com a nuvem.

Combinando diversas tecnologias, foi possível criar novos padrões de segurança. Uma estratégia de grande importância é a *Data Loss Prevention* (Prevenção de Perda de Dados). Com essa estratégia é possível diminuir consideravelmente as chances de acontecer um vazamento de informações confidenciais. Esta estratégia auxilia na detecção de tráfego na rede, e é capaz de bloquear remotamente os dados sensíveis. Assim é possível evitar que um *smartphone*, por exemplo, vaze dados sensíveis da corporação. A segurança antiga de rede não é mais suficiente e é preciso criar novas estratégias combinando diferentes sistemas DLP (EARLS, 2015).

2.2 PROBLEMAS DE SEGURANÇA

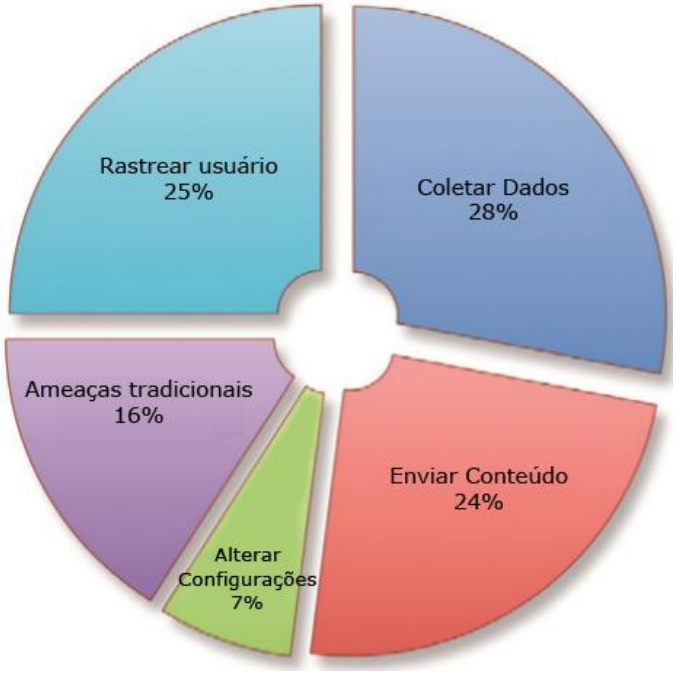
A adoção do modelo BYOD pelas corporações é considerada muito positiva. Os empregados têm maior rendimento no seu trabalho diário por utilizarem produtos anteriormente conhecidos. Porém os usuários, em suma, não tem conhecimento suficiente para identificar possíveis falhas de segurança em seus dispositivos. Este é o grande desafio da gerência de TI quando usufruem do BYOD.

De acordo com o *International Journal in IT and Engineering* (BOATEN e OSEI, 2016), existem quatro grandes categorias de vulnerabilidades BYOD. São elas: vulnerabilidades associadas a *malwares*, vulnerabilidades de permissão de usuário, vulnerabilidades de encriptação e a vulnerabilidade do usuário desatento.

Conforme a pesquisa citada anteriormente, realizada pela *Harris Interactive* e ESSET (MORROW, 2012), 47% dos funcionários utilizam o *desktop* pessoal para acessar dados corporativos, enquanto 41% fazem isso através de *notebooks*, 24% com *smartphones* e 10% com *tablets*. No entanto, menos da metade desses dispositivos são protegidos por sistemas básicos de segurança. O potencial para ataques cibernéticos, neste caso, é maior, pois permite a utilização de *keyloggers*, *malwares* e outros, através de dispositivos *endpoint*.

De grande popularidade, *malwares* são *softwares* maliciosos desenvolvidos por *hackers* com diversos objetivos finais (ISACA, 2016). Dentro desta definição é possível destacar os famosos cavalos de Tróia, os *adwares* e os *spywares*. Cavalos de Tróia são programas que têm como finalidade criar uma porta de entrada para facilitar invasões. *Adwares* são responsáveis por exibir grandes quantidades de propagandas, sem a permissão do usuário, baseando-se em seus gostos, através das buscas e compras recentes. *Spywares* recolhem diversas informações do usuário, como por exemplo, tudo que ele digita e acessa na internet, sem que ele saiba disso. A Figura 3 demonstra as proporções destas ferramentas e o que elas fazem, apenas no segmento de *smartphones*.

Figura 3 - O que um malware faz com seu smartphone.



Fonte: SYMANTEC, 2011.

Em 2016, a venda diária de *smartphones* chegou ao patamar de 3.8 milhões de unidades ao redor do mundo (COMPUTER WORLD, 2016). Com tamanha quantidade de dispositivos sendo comercializados, torna-se óbvio o interesse dos *hackers* na utilização desta plataforma para coletar e roubar dados pessoais e empresariais.

A criação de novos *malwares* é constante e seu principal foco é o Android, pois é o sistema mais comum atualmente. Um funcionário descuidado pode vazar informação confidencial apenas por ter acessado a rede corporativa e salvo algum documento em seu armazenamento local. Um *malware*, por exemplo, pode invadir o cartão SD e roubar o arquivo facilmente (MORROW, 2012).

Além da rede *Wi-Fi*, o *Bluetooth* pode ser utilizado pelo empregado durante sua jornada de trabalho. Basicamente, o *Bluetooth* funciona através do pareamento entre dispositivos. Este pareamento é feito através de uma chave que é utilizada na criptografia dos dados transmitidos. A chave é um dos principais alvos de ataques, pois, caso o hacker tenha conhecimento dela, ele poderá ter acesso a toda informação trocada. Existem diversas fraquezas que podem ser exploradas, por exemplo, ataques de negação de serviço. Mais informações são citadas no livro *Bluetooth Security Attacks: Comparative Analysis, Attacks*

and Countermeasures de HAATAJA, K., HYPPÖNEN, K., PASANEN, S., TOIVANEN, P. (2013).

Por fim, existe o risco humano que precisa ser calculado pela corporação. Um empregado mal intencionado pode facilmente roubar segredos de mercado, projetos, propriedade intelectual ou informações sensíveis, salvando estes dados no seu dispositivo ou em sua conta na nuvem. Já um empregado que não possua o mínimo conhecimento da área, pode facilitar o acesso a terceiros, ou, inclusive, realizar o vazamento de dados sensíveis, sem perceber. Por isso, a empresa precisa treinar seus funcionários e ainda assim controlar os dados, não só dentro da sua infraestrutura, como também no dispositivo final para prevenir vazamentos, acidentais ou intencionais, das informações corporativas (MORROW, 2012).

2.3 ESTRATÉGIAS DE MOBILIDADE

Existem quatro conceitos estratégicos que podem auxiliar a modernização de uma corporação. Além do já detalhado BYOD, é possível citar o HYOD (*Here is Your Own Device*), o CYOD (*Choose Your Own Device*) e o OYOD (*On Your Own Device*). Todos eles são focados na utilização de dispositivos móveis, mas com diferentes formas de administração. Assim, a empresa pode escolher qual método de segurança mais lhe agrada (JGRCS, 2010).

O conceito HYOD (*Here is Your Own Device* - Aqui está o Seu Dispositivo) é caracterizado pelo fato da corporação fornecer o dispositivo móvel. Deste modo a empresa tem controle total do dispositivo em questão. Ela também vai fornecer suporte completo ao dispositivo, desde instalação, configuração e outros.

O modelo CYOD (*Choose Your Own Device* - Escolha o Seu Dispositivo) permite que a corporação determine previamente alguns modelos de dispositivos para servir de opção ao funcionário. Deste modo as políticas de segurança não são tão rígidas e o usuário tem permissão para instalar programas específicos.

Já no OYOD (*On Your Own Device* - Em Seu Dispositivo), o usuário final, ou seja, o funcionário pode utilizar qualquer tipo de dispositivo móvel pessoal. Porém, ele mesmo é responsável pelo total gerenciamento do dispositivo e segurança dos dados (corporativos e pessoais) ali armazenados. Diferentemente do BYOD, a empresa não utiliza e não determina previamente nenhuma espécie de políticas de segurança.

Figura 4 - Comparativo: Estratégias Móveis e Controle Corporativo.

Fonte: JGRCS, 2010.

De acordo com a pesquisa de JGRCS (2010), o colaborador prefere a utilização do seu próprio dispositivo, como ilustrado na Figura 4. Desta forma ele tem mais liberdade e consequentemente mais satisfação ao trabalhar. No entanto, as corporações tendem a optar por um equilíbrio entre a liberdade individual e a segurança da informação. A existência de ameaças virtuais é uma dura realidade e o empresário precisa estar ciente dos riscos inclusos em cada conceito de mobilidade. Uma vez que a empresa adote o modelo BOYD, ela vai possibilitar um meio termo entre satisfação do funcionário e controle de dados sensíveis. Ao trabalhar de maneira conjunta, tanto o funcionário quanto o empresário terão maior produtividade caso utilizem corretamente as ferramentas atuais.

2.4 CONSIDERAÇÕES FINAIS

BYOD é o conceito capaz de permitir o uso da tecnologia móvel, que está em constante evolução, no ramo profissional. Existem diversas tecnologias de informação trabalhando em conjunto, formando o cenário necessário para que o colaborador possa acessar os dados corporativos de qualquer lugar e em qualquer dispositivo. Porém algumas medidas básicas de segurança precisam ser adotadas. A organização precisa estar ciente dos riscos que cada estratégia móvel pode representar. Além disso, o funcionário também deve ter uma ciência básica de como utilizar seu equipamento.

Diante das vulnerabilidades relatadas, é necessária uma reorganização no setor de TI da empresa que recém adotou este conceito. Novas ferramentas de segurança serão indispensáveis para que o funcionamento interno continue sem problemas, e para tanto, alguns hábitos pessoais dos funcionários precisam mudar. Uma vez que o dispositivo móvel pessoal é usado também para o trabalho, é indispensável mais atenção no uso, além de evitar sites e/ou programas suspeitos. O uso do BYOD tende a crescer mundialmente. Sendo assim, novos métodos de segurança serão criados. Tanto o colaborador quanto a corporação deverão estar dispostos ao novo conceito de trabalho e também atentos aos seus pontos negativos.

Atualmente, inúmeras ferramentas de administração e gerenciamento de redes BYOD estão disponíveis no mercado. Outras tantas estão sendo desenvolvidas e testadas por diversas empresas. Estes utilitários são capazes de proporcionar maior controle e segurança. Algumas delas serão apresentadas no próximo capítulo.

3. CARACTERIZAÇÃO DA PESQUISA

O capítulo de caracterização da pesquisa tem o intuito de exibir a metodologia de pesquisa que foi utilizada e quais as tecnologias, ou seja, as ferramentas voltadas para o modelo BYOD que estão disponíveis no mercado. Estas ferramentas têm diversos objetivos finais. Porém, todas podem ser utilizadas durante a administração e monitoramento de segurança da corporação que adotar o modelo BYOD. O processo de análise e desenvolvimento do protótipo foi realizado de acordo com as informações descritas neste capítulo.

3.1 METODOLOGIA DE PESQUISA

A coleta de informações relacionadas ao modelo BYOD foi feita através de pesquisas em artigos, *ebooks*, revistas eletrônicas, páginas web e livros. É possível classificar como exploratória a pesquisa realizada sobre ferramentas para administração e segurança BYOD. Este tipo de pesquisa tem como objetivo proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a construir hipóteses (GEHARDT, SILVEIRA, 2009). Uma vez que o objetivo final deste projeto é propor um protótipo de software, a pesquisa exploratória é a opção mais adequada.

É interessante destacar que a utilização de um protótipo pode revelar diversas informações que até então a pesquisa não tinha capacidade de explorar. Por exemplo, um profissional de TI que utiliza uma ferramenta de administração BYOD pode seguir determinados processos. Uma vez que ele conheça um novo protótipo com mais funcionalidades e praticidades, suas opiniões podem mudar. Este profissional ainda pode acabar influenciando a forma como a organização administra sua rede e seus dados sensíveis. Supondo que o protótipo seja superior à ferramenta utilizada, a empresa pode se tornar um cliente em potencial e consequentemente transformar economicamente o protótipo em uma ferramenta viável.

A tecnologia que permite o uso do BYOD é relativamente nova. Por se tratar de um conceito recente, a quantidade de documentação científica é um pouco restrita. Além disso, o BYOD não é um assunto tão trivial ao público leigo em geral quanto outras tecnologias, como por exemplo, a computação em nuvem. Um dos objetivos da pesquisa realizada é popularizar

a filosofia de trabalho com dispositivos móveis. Isso caracteriza a definição, citada anteriormente, da pesquisa exploratória.

3.2 FERRAMENTAS PARA GERENCIAMENTO BYOD

Para que o modelo BYOD funcione de maneira adequada e segura, algumas técnicas de administração e segurança precisam ser adotadas. Cada organização tem uma maneira própria de lidar com isso. Enquanto algumas organizações são mais rigorosas e utilizam maiores níveis de segurança, outras são mais simples e não adotam grandes medidas administrativas. Entretanto, independentemente da rigorosidade que as corporações escolherem seguir, elas vão usufruir de alguns *softwares* específicos. Estes *softwares* foram tratados neste capítulo como ferramentas.

As ferramentas BYOD foram projetadas para auxiliar os profissionais de TI responsáveis por manter a rede e os dados sensíveis da empresa seguros e em perfeito funcionamento. Com o intuito de apresentar tais ferramentas e suas funcionalidades, inúmeras buscas foram realizadas. Inicialmente, a pesquisa direcionou-se para artigos e periódicos relacionados com o BYOD. Dentro desta abordagem é possível destacar a grande quantidade de alertas de segurança que os especialistas publicam frequentemente. Estudos mostram que a adoção do BYOD cresce juntamente com a quantidade de problemas de segurança. A maioria destes artigos indicam que só a utilização de ferramentas não é suficiente. Estratégias de segurança envolvendo o treinamento dos funcionários são essenciais.

De acordo com a publicação de PwC (2015), três quartos das grandes corporações tiveram casos de violação de dados envolvendo funcionários em 2015. Além disso, aproximadamente 50% dos problemas de grande impacto foram causados por algum tipo de erro humano.

Devido à pouca quantidade ou até ausência de publicações científicas referenciando ferramentas, a pesquisa por *softwares* foi feita com base em sites de revistas de tecnologia e afins. Sites de colaboração entre desenvolvedores *open-source*, como o CodeProject[2] por exemplo, também foram de grande importância. Este tipo de plataforma permite que um usuário auxilie o outro, com opiniões e esclarecimento de dúvidas sobre programação. Esta opção de colaboração é utilizada pela maioria das ferramentas *free* e/ou *open-source*

[2] *CodeProject – For those who code. Free source code and tutorials for Software Developers and Architects* (CodeProject – Para aqueles que programam. Códigos fonte gratuitos e tutoriais para desenvolvedores e arquitetos de *software*). Mais informações em: <https://www.codeproject.com/>.

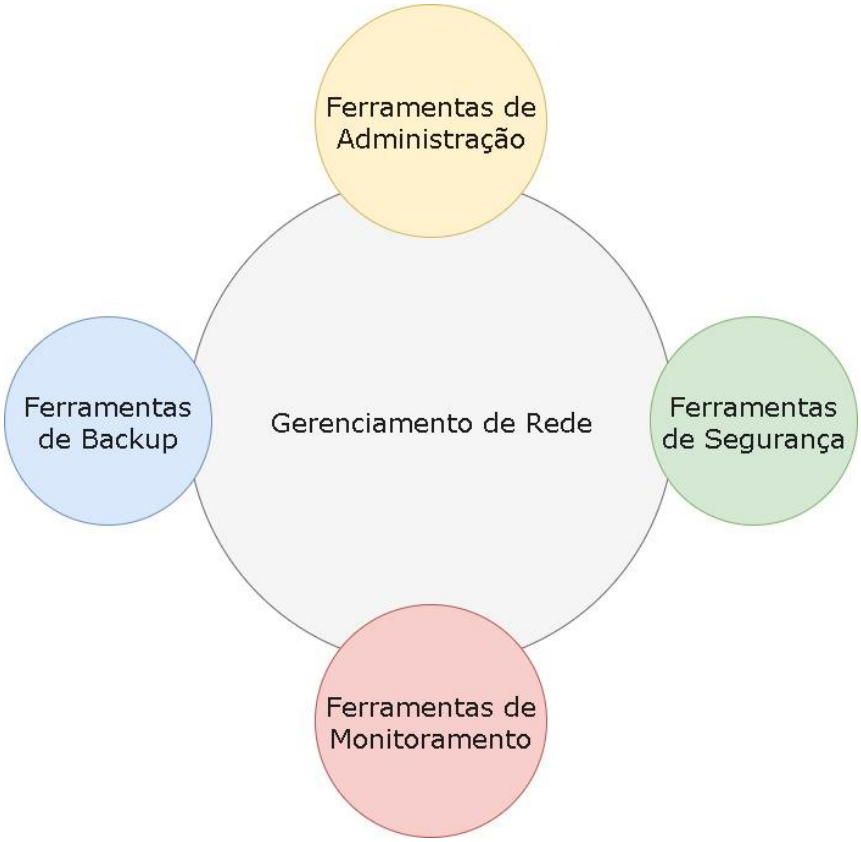
pesquisadas. Estas plataformas ganham cada vez mais usuários justamente pelas políticas de compartilhamento gratuito na internet. E por fim, foram realizadas pesquisas em sites de notícias especializados em informática e *blogs* que continham comparativos e opiniões pessoais dos respectivos autores.

Através destas fontes citadas anteriormente, foi possível detalhar quais as funções finais das ferramentas BYOD disponíveis e quais formas de licença de uso são empregadas por cada uma delas.

3.3 CLASSIFICAÇÕES E TIPAGEM DE FERRAMENTAS

Dentre as ferramentas de rede, podem-se destacar quatro principais classificações. São elas: ferramentas de administração, ferramentas de segurança, ferramentas de monitoramento e ferramentas de *backup.* Todas essas ferramentas trabalhando em conjunto compõem a suíte de gerenciamento de rede, ilustrada na Figura 5.

Figura 5 - Suíte de Gerenciamento de Rede.



Fonte: Autoria própria.

Dentro das ferramentas de administração é possível destacar as que realizam o levantamento de maquinário e análise de dados da rede. Estas ferramentas normalmente fornecem diversos tipos de relatórios. Através delas é possível saber, por exemplo, quantos dispositivos Android estão conectados na rede. De acordo com o catálogo de *software* da NASA (2017), existem diversas opções de ferramentas para medição de estatísticas e indicadores de integridade e uma delas é a STAT (*Schedule Test and Assessment Tool*).

As ferramentas de segurança são compostas por antivírus, *firewalls*, testadores de senhas, entre outros. Elas são de extrema importância para a corporação. De acordo com o site toolwatch.org [3] as melhores ferramentas de 2014 são as responsáveis por descobrir vulnerabilidades na rede e principalmente em aplicações web e navegadores. Esta lista de ferramentas contém um *software open-source* capaz de explorar vulnerabilidades presentes na plataforma Facebook. As vulnerabilidades do Facebook podem comprometer os dados da organização caso o funcionário utilize um dispositivo móvel com o aplicativo instalado e acesse informações sensíveis. Já o site youlinux.com [4] mostra um compilado de diversas ferramentas para Linux com os mais variados objetivos finais, incluindo até ferramentas para roubar informações. Além disso, o site também indica alguns livros essenciais de segurança de redes corporativas.

As ferramentas de monitoramento são responsáveis por mostrar o status em tempo real da rede. As mais comuns são os *sniffers* de rede e os softwares de controle de fluxo de rede. Ainda podem existir outras subdivisões. Em conformidade com PHATAK (2012) existem duas classificações de *scanners* de monitoramento: *network scanner* e *web scanners*. *Scanners* de rede são responsáveis pela constante análise dos pacotes que estão circulando na rede. Essa verificação precisa ser realizada nos dois lados do *firewall*. Já os *Web Scanners* precisam realizar uma verificação maior. Normalmente estas ferramentas analisam o intervalo entre as camadas de rede 2 (Camada de Enlace) e 7 (Camada de Aplicação).

As ferramentas de *backup* são responsáveis por manter a integridade dos dados em casos de falhas, que podem ser de sistema ou físicas. As corporações devem seguir os seguintes passos: desenvolver um bom plano de *backup*; gerenciar os *backups* de maneira efetiva; executar testes de recuperação periodicamente; ter *backup* e recuperação SLA; ter um

[3] A lista "*2014 Top Security Tools as Voted by ToolsWatch.org Readers*" está disponível em: <http://www.toolswatch.org/2015/01/2014-top-security-tools-as-voted-by-toolswatch-org-readers/>.

[4] O compilado de ferramentas para Linux está disponível em: <http://www.yolinux.com/TUTORIALS/LinuxSecurityTools.html>.

plano de recuperação DRP documentado; manter-se informado e estudando os sistemas operacionais e suas ferramentas de *backup* (ISACA JOURNAL, 2012).

Existem inúmeras ferramentas disponíveis no mercado, algumas opções gratuitas, outras pagas. Configurar o funcionamento em paralelo destas ferramentas pode resultar em um pacote completo de gerenciamento de rede. Este gerenciamento pode ser mais simples ou mais rigoroso, variando de acordo com as políticas de segurança que a corporação definiu previamente.

O mercado de ferramentas com foco no modelo BYOD é um pouco mais restrito, entretanto é bem atendido. Existem ferramentas de todas as classes que funcionam normalmente em diversos sistemas operacionais, ou seja, elas funcionam em ambientes compostos por diversos tipos de dispositivos.

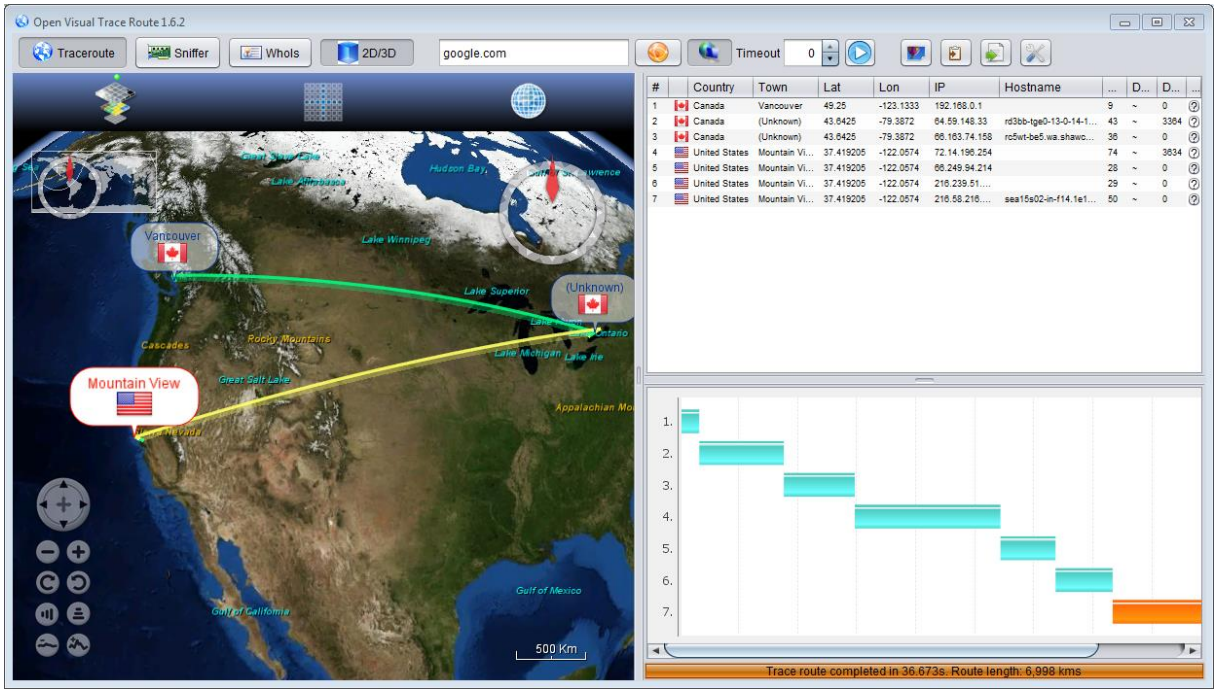
3.4 EXEMPLOS DE FERRAMENTAS DISPONÍVEIS NO MERCADO

Algumas ferramentas que atendem o modelo BYOD e estão disponíveis no mercado são apresentadas nas próximas duas seções. Estas ferramentas são licenciadas de diferentes formas. Com o objetivo de explorar as diversas finalidades que as ferramentas propõem, foi realizada uma descrição citando também qual o tipo de licença de uso adotado. A descrição da ferramenta contém também uma justificativa referente à inclusão ou exclusão da mesma no projeto de protótipo de integração proposto. Durante a pesquisa, algumas ferramentas se destacam, tais como: Open Visual Traceroute, OpenVAS, Snort, Ntop, Cain & Abel e a Mobi Control.

3.4.1 OPEN VISUAL TRACEROUTE

O *Open Visual Traceroute* é uma ferramenta desenvolvida em Java. Classificada como um *software open-source* e também como *cross plataform* (solução multiplataforma), ou seja, é capaz de funcionar normalmente em diversos sistemas operacionais e arquiteturas. Como o próprio nome sugere, o *Open Visual Traceroute* é capaz de apontar visualmente em um mapa-múndi, 3D ou 2D, o caminho pelo qual os pacotes de rede trafegam. A Figura 6 mostra uma das telas do programa, onde é possível identificar por quais longitudes, latitudes, cidades e países os dados estão passando.

Figura 6 - Demonstração do Open Visual Traceroute.



Fonte: VISUALTRACEROUTE, 2017.

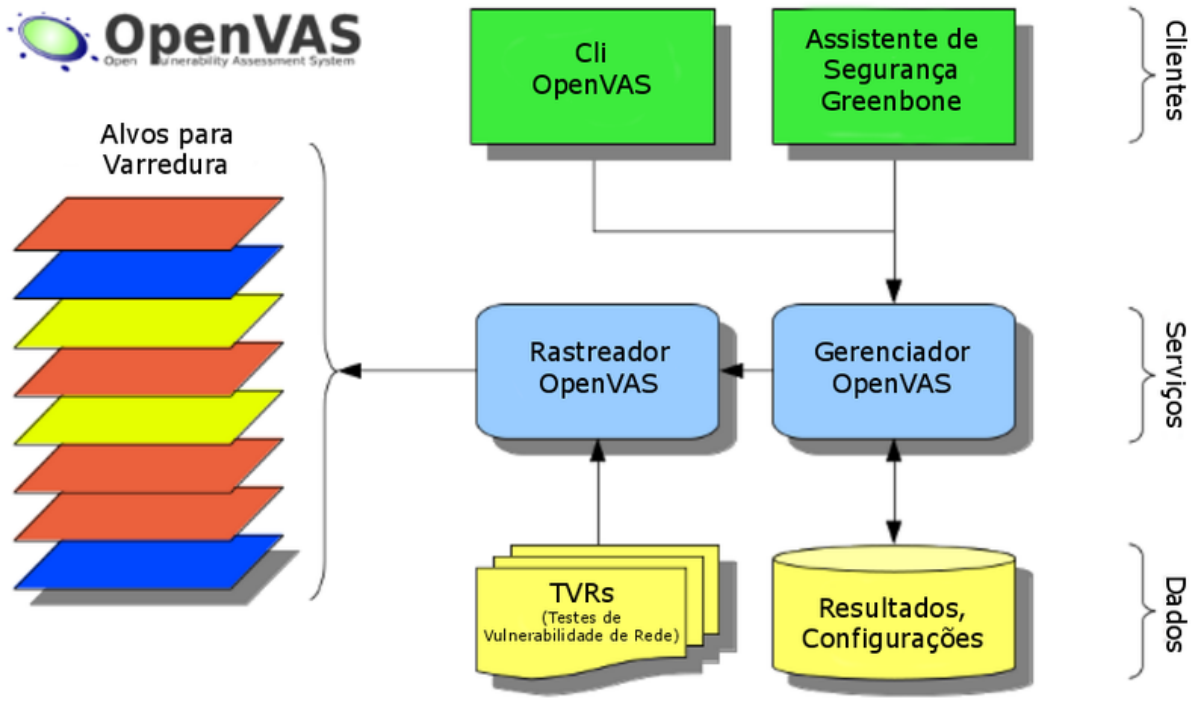
Plataformas mais simples podem optar pelo mapa 2D que exige menos recursos computacionais. Além disto, a ferramenta ainda inclui um *sniffer* de rede que permite explorar quais dados estão sendo trocados entre a rede local e a internet. Outro diferencial desta ferramenta é a aba "*Whois*" (Quem é?) capaz de entregar informações públicas sobre o domínio escolhido. Por exemplo, através do endereço IP 208.80.154.224, é possível identificar que o *hostname* é "text-b.eqiad.wikipedia.org", que ele foi criado no dia 16 de março de 2003 e foi registrado na cidade de São Francisco nos Estados Unidos (VISUALTRACEROUTE.NET, 2017).

3.4.2 OPENVAS

O *OpenVAS* é classificado como um *framework* composto de diversos serviços e ferramentas focados em proporcionar um completo gerenciamento de vulnerabilidades. Ele surgiu como uma alternativa após a mudança de licença, que passou a ser código fechado, da ferramenta de varredura *Nessus*. Esta ferramenta *open-source* é capaz de relatar quais vulnerabilidades estão presentes no *host* e ainda, através de *plug-ins*, sugerir possíveis soluções ao administrador. A interface é acessada através de navegadores web. Sua

arquitetura possui o *OpenVAS Scanner* (Rastreador OpenVAS), responsável pela execução constante dos TVRs (Testes de Vulnerabilidade de Rede). O gerenciador *OpenVAS* é o serviço central, ou seja, ele contém a inteligência do software e ainda controla os outros blocos, como por exemplo, o Rastreador *OpenVAS* (OPENVAS.ORG, 2017). A Figura 7 ilustra o funcionamento do *OpenVAS* através da sua relação estrutural, composta por 4 grandes ambientes. São eles: alvos para varredura, clientes, serviços e dados.

Figura 7 - Estrutura Funcional do OpenVAS.



Fonte: WIKIPEDIA, 2014.

O *OpenVAS* é listado como o quarto melhor *scanner* de segurança de rede de acordo com PHATAK (2012).

3.4.3 SNORT

O *Snort* é uma ferramenta *open-source* NIDS (*Network Intrusion Detection System*). Ela foi desenvolvida por Martin Roesh e se tornou bastante popular por sua flexibilidade nas configurações, além das constantes atualizações. Recentemente a *Cisco* adquiriu a *Sourcefire*, atual responsável pelo desenvolvimento da ferramenta (DE LA MERCED, 2013). Sendo

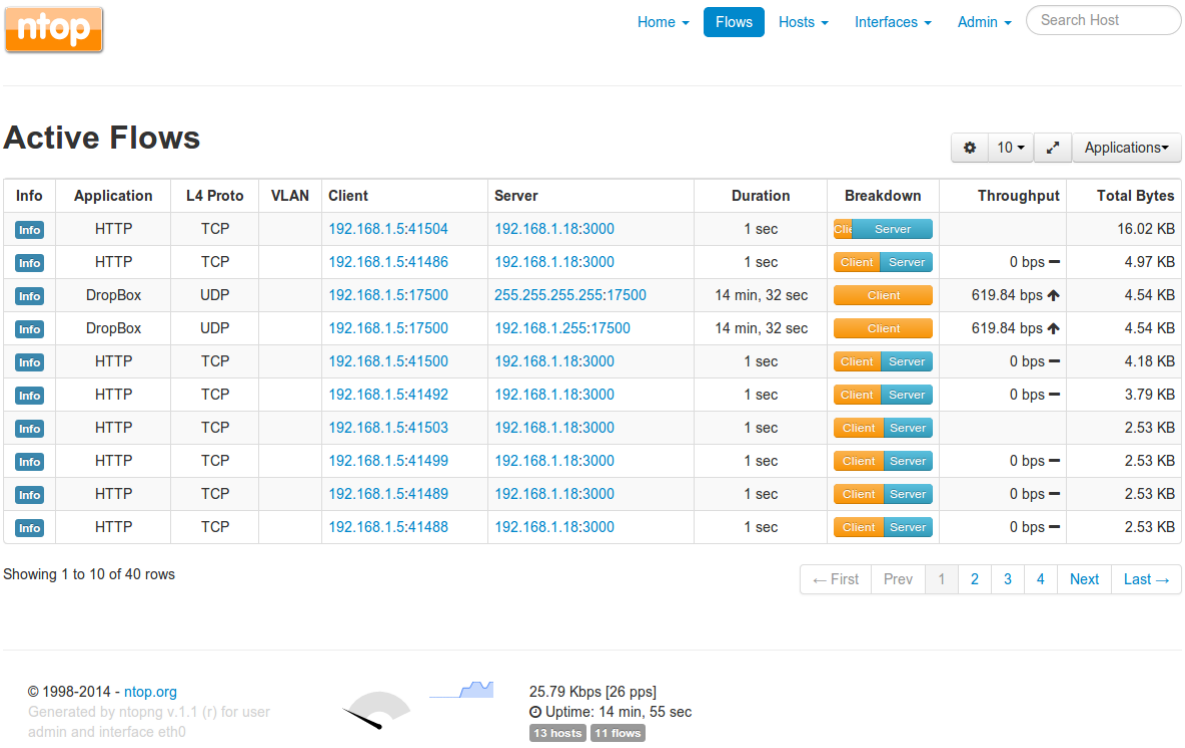
assim, a *Cisco* possui os direitos sobre este *software*. Esta ferramenta foi desenvolvida em linguagem C e é capaz de verificar, em tempo real, os mais variados tipos de problemas na rede com a qual o computador estiver conectado. Como o Snort não demanda grandes recursos de processamento, ele é indicado para monitoramento de redes TCP/IP de pequeno porte. Um dos requisitos para funcionamento é a utilização do banco de dados MySQL (MARTINEZ, 2015). Além disso, o Snort também é uma ferramenta *cross plataform*. Ela é executável em arquiteturas CISC (*Complex Instruction Set Computer*) e RISC (*Reduced Instruction Set Computer*) e em diversas distribuições de sistemas operacionais.

Apesar de ser um programa que mostra seus resultados em console, é possível encontrar diversas opções de GUIs (*Graphical User Interface*) para ele, sendo uma delas projetada pela própria *Cisco* (ESLER, 2011). Por fim, a ferramenta Snort pode ser configurada em três principais modos: *sniffer*, *logger* de pacotes e detector de intrusão. No modo *sniffer*, o Snort irá interceptar os pacotes que estão circulando na rede e mostrá-los no console. No modo *logger* de pacotes, os pacotes de rede serão gravados no disco. No modo detector de intrusão, o tráfego de rede será constantemente monitorado e analisado perante um conjunto de regras de segurança, que foram anteriormente definidas pela corporação ou pelo administrador de rede (SNORT.ORG, 2017).

3.4.4 NTOP

Ntop, ou Ntopng, é uma ferramenta *open-source*, capaz de ser executada em diversos sistemas operacionais e plataformas, como por exemplo, distribuições Linux e Windows, além de arquiteturas x86 e ARM. O Ntop é um *sniffer* de pacotes de rede com uma interface web. Ele é baseado na biblioteca "*libpcap*". Sendo que uma de suas vantagens é o uso reduzido de processamento. Mas é importante destacar que quanto maior a rede de computadores em que ele será utilizado, maior será o requisito de processamento (PINHEIRO, 2014). Ele foi desenvolvido por Luca Deri, um pesquisador e administrador de rede da Universidade de Pisa, na Itália. A ferramenta utiliza o sistema *Round-Robin Database Tool* para armazenar as estatísticas de tráfego. As estatísticas de rede são geradas através do monitoramento dos seguintes protocolos: TCP/UDP/ICMP, ARP, IPX, DLC, DECnet, *AppleTalk* e *Netbios* (NTOP.ORG, 2017). A Figura 8 mostra como a interface web do Ntop facilita a visualização dos dados pelo usuário.

Figura 8 - Interface Web do Ntop.



Fonte: MUNDOTIBRASIL, 2014.

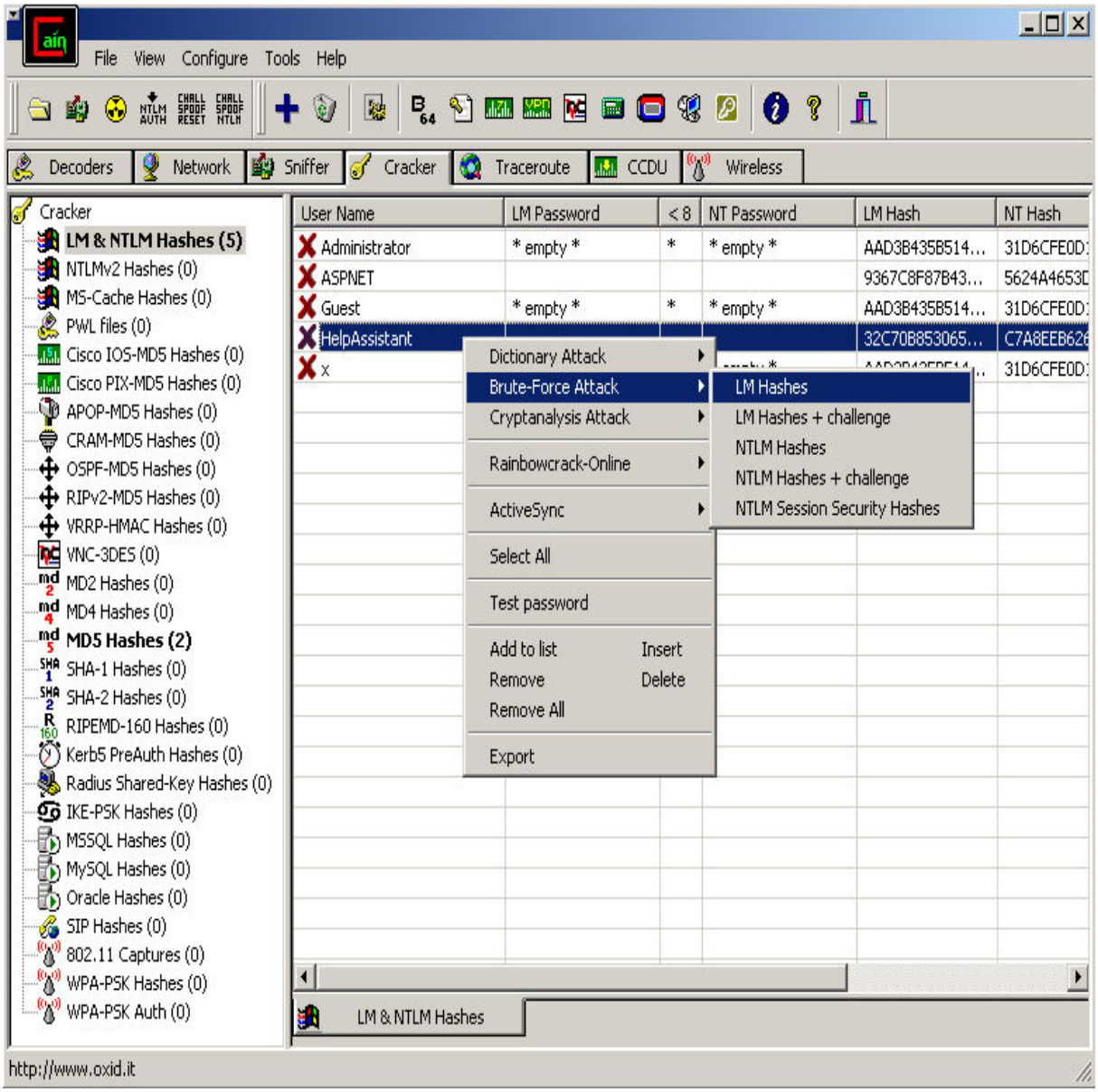
O Ntop foi indicado como uma das sete melhores ferramentas *open-source* de monitoramento de rede pela revista InfoWorld (2014).

3.4.5 CAIN & ABEL

A ferramenta de recuperação de senha Cain & Abel é um *software* que funciona no ambiente Windows. Este projeto foi desenvolvido por Massimiliano Montoro e conta com diferentes métodos úteis para a quebra de senhas. Cain & Abel é capaz de revelar senhas armazenadas em máquinas com Windows instalado. Além disso, o *software* pode descobrir senhas MySQL e algumas senhas de dispositivos *Cisco*, entre outras diversas opções (SKOUDIS, 2013). O usuário pode escolher entre alguns algoritmos, como por exemplo, o de força bruta. A utilização de um amplo dicionário com palavras-chave também pode ser escolhido durante o processo de quebra de senha (MOHAMED, 2013). A corporação pode fazer uso deste tipo de solução com o intuito de garantir um nível mínimo de segurança para as senhas de seus colaboradores. Montoro (2014) ainda deixa claro que não apoia nenhum tipo de uso ilegal de sua ferramenta. Ela foi desenvolvida com o intuito de ajudar

profissionais de segurança de TI. A Figura 9 mostra a interface de usuário e as opções de algoritmos disponíveis, além de relacionar as senhas armazenadas no sistema.

Figura 9 - Interface de Usuário Cain & Abel.



Fonte: SECTOOLS, 2014.

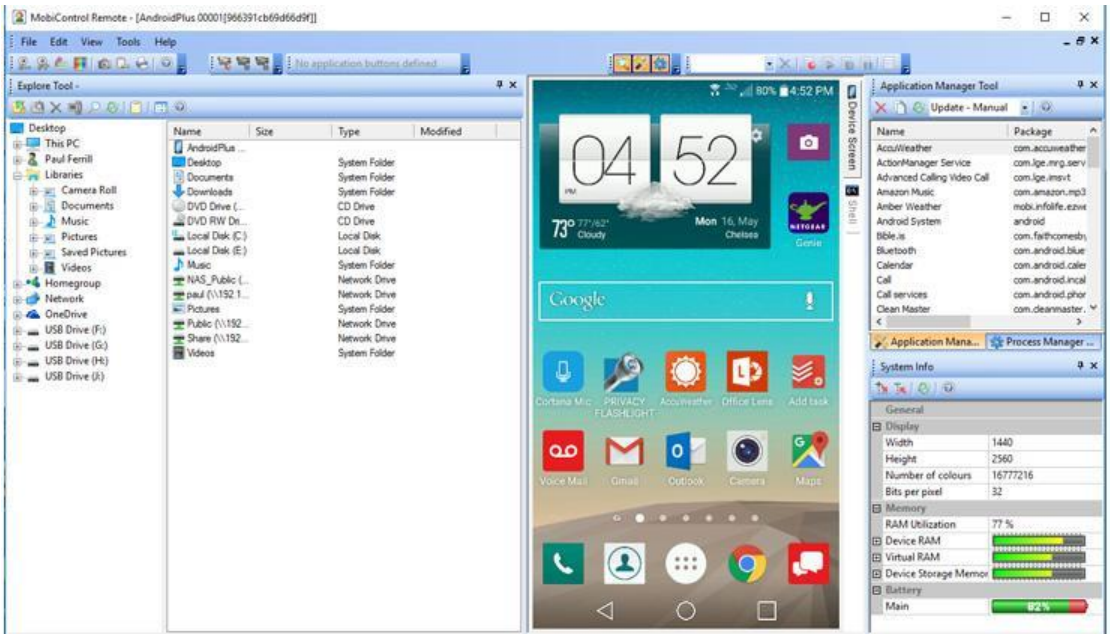
A ferramenta Cain & Abel é relacionada entre as cinco melhores ferramentas *hacker* por SKOUDIS (2013) e indicada pelo site *sectools.org*.

3.4.6 MOBI CONTROL

A ferramenta Mobi Control é a solução EMM (*Enterprise Mobility Management*) desenvolvida pela SOTI. Esta solução é planejada justamente para empresas que adotam o modelo BYOD. Mobi Control é desenvolvido para diversas plataformas móveis com o objetivo de facilitar o gerenciamento e segurança dos dados corporativos que podem estar circulando por estes dispositivos. Ela propõe uma suíte completa de gerenciamento remoto, incluindo relatórios gráficos e diversas análises. Além disso, oferece um serviço completo de segurança e geolocalização de dispositivos. Entre as corporações que utilizam esta solução, é possível destacar McDonald`s, Nike, Coca-Cola, Fedex e até empresas de tecnologia como a HP. O Mobi Control tem mais de 12.000 implantações corporativas e milhões de dispositivos móveis gerenciados globalmente (SOTI.NET, 2017). Os administradores de TI podem realizar suporte completo aos dispositivos, de forma remota. Existem diversas empresas no Brasil que realizam a implementação e configuração desta ferramenta em outras corporações que estiverem interessadas.

FERRIL (2016), colunista da revista PCMAG, realizou uma análise da ferramenta e classificou-a como excelente. A Figura 10 destaca a interface de gerenciamento remoto. Na imagem um dispositivo Android é acessado remotamente.

Figura 10 - Gerenciamento Remoto de Dispositivo Android pelo Mobi Control.



Fonte: FERRIL, 2016.

É necessário destacar que os custos mensais do Mobi Control começam em torno de quatro dólares por dispositivo gerenciado.

3.4.7 AIRCRACK-NG

Uma ferramenta interessante que atende completamente o gerenciamento de redes sem fio corporativas que utilizam BYOD é a Aircrack-ng. Esta ferramenta pode ajudar o profissional de TI a melhorar a segurança de seu ambiente antes que outra pessoa mal intencionada lhe cause sérios problemas. Senhas de redes sem fio que utilizam padrões de segurança como WEP, WPA ou WPA2 podem representar uma grande vulnerabilidade à segurança da rede de computadores da corporação (RUBENS, 2007).

Aircrack-ng é um conjunto de aplicativos capaz de identificar todos os *access-points* presentes na rede, checar se a rede está protegida com sistemas de criptografia de dados e ainda testar a força das chaves de acesso. Esta ferramenta funciona com qualquer placa de rede sem fio 802.11 cujo driver suporte o modo de monitoramento bruto. Uma lista completa de hardwares compatíveis está disponível no site da ferramenta. O Aircrack-ng possibilita que uma chave WEP possa ser quebrada em apenas alguns minutos. Dentro do Aircrack-ng, é possível destacar o Airmon-ng, ferramenta que faz com que o adaptador wireless receba todo o tráfego de rede sem fio, incluindo pacotes direcionados para outros endereços MAC. Uma vez que o Airmon-ng esteja habilitado, é possível combinar sua utilização com o Airodump-ng. O Airodump-ng permite que sejam capturados pacotes específicos da rede. Estes pacotes podem conter dados cruciais de uma rede sem fio.

A Figura 11 mostra uma relação de *access points* com seus respectivos endereços MAC, o canal utilizado, a velocidade, o método de encriptação, seu respectivo nome (ESSID), entre outras informações. Todas essas informações foram obtidas utilizando o Airodump-ng.

Todas as ferramentas contidas nesta suíte funcionam através de linhas de comando. Ela foi desenvolvida primeiramente para distribuições Linux, mas também funciona no Windows, Mac OS X e outros. A última versão da suíte foi lançada em 2016, e ela é desenvolvida por Thomas d'Otreppe (AIRCRACK-NG.ORG, 2017). Além disso, esta ferramenta faz parte da maior distribuição Linux voltada para segurança de redes, o Kali Linux (KALI.ORG, 2014).

Figura 11 - Resultados da ferramenta Airodump-ng.

Fonte: NULL BYTE, 2013.

3.4.8 CONSIDERAÇÕES SOBRE AS FERRAMENTAS

Em função da dificuldade de encontrar ferramentas que fossem compatíveis com a ideia proposta, os softwares pagos também foram citados na pesquisa. Estas sete ferramentas foram apresentas por serem desenvolvidas e planejadas visando à segurança ou ao gerenciamento de redes corporativas que permitem a utilização do padrão BYOD. Algumas delas, como o Snort e o OpenVAS são bem semelhantes, porém cada uma possui uma maneira diferente de trabalhar, ou seja, uma arquitetura de *software* diferente. Ainda são notáveis as diferentes formas de apresentar os resultados obtidos. Algumas destas soluções possuem apenas um console de comandos, outras possuem interface de usuário *desktop* e outras são ferramentas de interface web. Uma vez que a corporação tenha definido quais os níveis de segurança adotados, as ferramentas devem ser estudadas para saber se atendem ou não as demandas. Outro ponto importante que a empresa deve estar ciente é a relação custo-benefício. Dentre as ferramentas BYOD apresentadas, algumas são gratuitas e outras são pagas.

Apesar da indiscutível finalidade e produtividade de cada uma das ferramentas apresentadas, a utilização de todas neste projeto acaba sendo inviável. Algumas opções são de código fechado, como por exemplo, o Mobi Control. Outras opções são complexas de serem configuradas, voltadas para grandes corporações mundiais e talvez não sejam compatíveis com o ambiente proposto nesta solução, uma vez que o protótipo busca realizar uma boa integração de ferramentas *open-source* numa única interface gráfica de usuário *desktop*.

A ferramenta Aircrack-ng, num primeiro momento, havia sido selecionada para compor a Seção 3.5 (Ferramentas *Open-Source*), porém ela foi removida por motivo de compatibilidade. No momento em que a Seção 4.4 (Definições de Tecnologia) estava sendo desenvolvida, foi possível detectar problemas que inviabilizariam a utilização da mesma neste projeto. Como citado no site oficial[5], o Aircrack-ng é uma ferramenta que depende de compatibilidade de *hardware*, *drivers* e ainda necessita de permissões que o ambiente Windows não disponibiliza.

O escopo deste projeto ainda propõe o foco principal em ferramentas que possuam uma finalidade um pouco mais relacionada para com a administração e segurança interna da rede corporativa. Deste modo, ferramentas ao estilo da *Open Visual Traceroute* acabam fugindo do assunto proposto nesta pesquisa exploratória.

3.5 FERRAMENTAS OPEN-SOURCE

Apesar da relação de ferramentas citadas anteriormente nesta pesquisa, o projeto ainda precisa de uma solução viável. A busca por artigos, indicações de *softwares* e *reviews* resultarou em novas opções de ferramentas. Visando tornar o objetivo final possível, as seguintes ferramentas *open-source* foram selecionadas: NetworkMiner, jNetMap, Universal Password Manager, Password Strength Meter, NetCalculator e IP Monitor. Todas elas são desenvolvidas com foco em segurança de redes BYOD.

Um breve resumo comparativo com informações básicas destas ferramentas pode ser conferido na Tabela 2 e os detalhes das mesmas são apresentados nas próximas seções.

[5] Conforme informação dos desenvolvedores do Aircrack-ng, existem diversas limitações presentes no ambiente Windows. Por exemplo, poucas placas wireless de *notebooks* tem suporte; no Windows só é possível realizar a captura de pacotes passivamente, ou seja, pode levar dias, semanas, meses ou eternamente para capturar pacotes suficientes para quebrar a chave WEP. Mais detalhes estão disponíveis em: < https://www.aircrack-ng.org/doku.php?id=portugues_pacote_aircrack-ng_no_windows_para_leigos>.

Tabela 2 - Ferramentas Selecionadas para o Protótipo.

Nome da Ferramenta	Funcionalidade	Plataformas Disponíveis	Site Oficial
Universal Password Manager	Centralizar usuários e senhas em uma só base criptografada.	Windows, Linux, Mac, Android.	http://upm.sourceforge.net/
Password Strength Meter	Medidor de força de senhas com base em algoritmos de força bruta.	Windows, Linux, Mac.	https://github.com/ericmdw/java-pwdstrength
NetCalculator	Calculadora de endereços e subredes ipv4 (Classes A B C D E).	Windows.	https://www.codeproject.com/Articles/11063/NetCalculator
NetworkMiner	Sniffer de rede.	Windows, Linux, Mac, FreeBSD.	http://www.netresec.com/?page=NetworkMiner
jNetMap	Representação gráfica da topologia de rede.	Windows, Linux, Mac.	http://www.rakudave.ch/jnetmap/?file=introduction
IP Monitor	Monitoramento do endereço de IP público.	Windows, Linux, Mac.	https://github.com/pupi1985/IPMonitor

Fonte: Autoria própria.

3.5.1 UNIVERSAL PASSWORD MANAGER

O *Universal Password Manager*, ou UPM, é uma ferramenta bem simples. Este *software* de código aberto permite que o usuário armazene todas as suas credenciais de usuário e senhas em uma só base de dados criptografada. Os dados ficam acessíveis através de uma única senha global. Seu diferencial em relação aos outros gerenciadores de senha *open-source* é a execução multiplataforma, a simplicidade e o sincronismo da base de dados entre os dispositivos. A base de dados pode ser sincronizada via HTTP ou via *DropBox*.

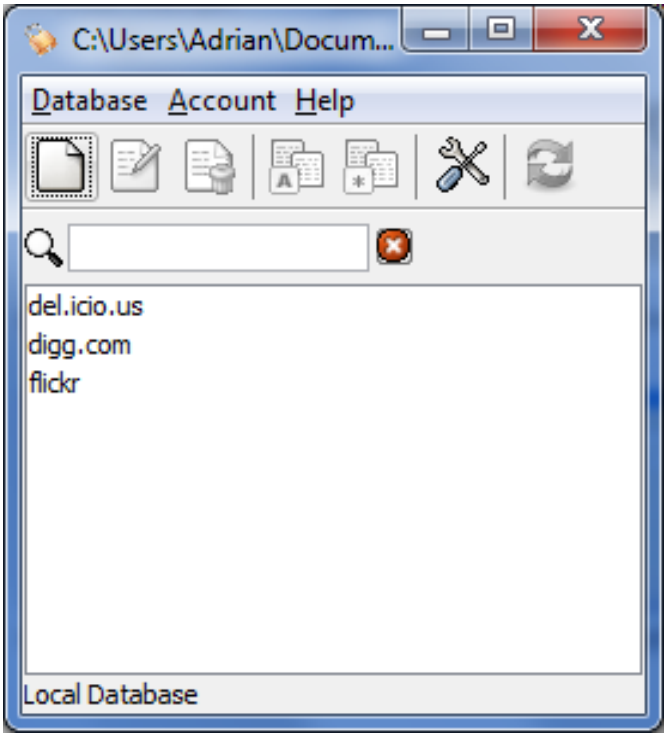
A criptografia da base de dados do *Universal Password Manager* é realizada seguindo o padrão AES[6] (*Advanced Encryption Standard* – Padrão de Criptografia Avançada). Para garantir a segurança das informações uma chave de 128 bits é utilizada. Para realizar o sincronismo da base de dados entre múltiplas máquinas via *DropBox*, basta salvar o arquivo no seu diretório pessoal na nuvem. Neste caso o *DropBox* garante a sincronização da base entre os dispositivos. Caso o usuário prefira sincronizar a base de dados via HTTP, ele

[6] O AES é um algoritmo de criptografia que opera com blocos de tamanho fixo de 16 bytes. Este padrão de criptografia pode trabalhar com chaves de 128, 192 e 256 bits. Uma explicação bem detalhada do padrão de criptografia AES pode ser acessada em: <https://pt.stackoverflow.com/questions/43492/como-funciona-o-algoritmo-de-criptografia-aes>.

precisará configurar um *web server* próprio que seja capaz de executar *scripts* em PHP. O *web server* deverá ser acessível de todos os dispositivos dos quais o usuário deseja sincronizar a base de dados. A configuração do *web server* deverá ser feita através do uso de ferramentas de terceiros pois o Universal Password Manager não oferece este serviço.

Ele tem versões compatíveis com ambientes Windows, Mac OS X, Linux e Android. Sua última atualização foi feita em 21 de março de 2016 (UNIVERSAL PASSWORD MANAGER, 2017). A Figura 12 mostra a interface simples do programa na versão Windows.

Figura 12 - UPM em Ambiente Windows.



Fonte: UNIVERSAL PASSWORD MANAGER, 2017.

3.5.2 PASSWORD STRENGTH METER

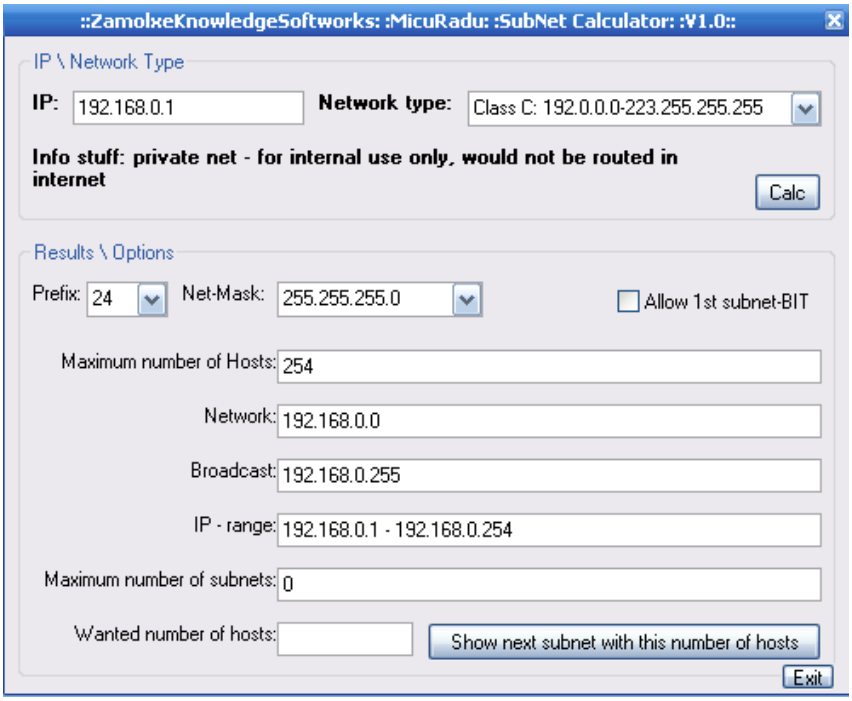
Outra ferramenta interessante relacionada com senhas é a *Password Strength Meter*. A *Password Strength Meter* é uma biblioteca Java desenvolvida por Eric Montgomery. Esta solução facilita a determinação e classificação da complexidade de uma senha de acesso. As classificações são feitas de acordo com os caracteres presentes na senha. Por exemplo, a senha "viniperini" é composta apenas por dez caracteres minúsculos e seguindo a lógica do código fonte seria classificada como "LENGTH_10_LOWER_CASE". Já a senha "AbCd123!" seria

classificada como “LENGTH_8_MIXED_CASE_WITH_NUMBER_AND_SYMBOL”. Ao classificar a senha pela sua complexidade, você pode substituir caracteres maiúsculos, minúsculos e alfanuméricos por combinações mais seguras. Por exemplo, a senha "ئ؏KjΩ☮" é mais resistente a ataques de força bruta do que a senha "aJ@2fz0z" apesar de não atender aos requisitos mínimos da maioria dos sistemas (MONTGOMERY, 2010). A biblioteca ainda fornece um cálculo aproximado de quantas iterações seriam necessárias para que um sistema comum de força bruta quebre a senha testada.

3.5.3 NETCALCULATOR

A ferramanta *NetCalculator* tem como finalidade principal agilizar a configuração de uma subrede. De maneira prática e extremamente rápida, o administrador de TI pode calcular endereços IP utilizando o *NetCalculator*. Cálculos de endereçamento de subredes não são complexos, mas podem tomar um tempo útil considerável em alguns casos. Essa ferramenta é capaz de calcular as diferentes classes de endereços IP. Um endereço IP é formado de um endereço “Rede” e um endereço “*Host*”. As classes são designadas conforme a quantidade de bits presentes no cabeçalho “Rede”. As classes A, B e C são conhecidas como classes primárias. A classe D é utilizada para *multicasting*, pois permite a entrega de pacotes a um conjunto de máquinas. A classe E não é utilizada atualmente (GALVÃO, 2016). A ferramenta ainda é capaz de informar a máscara de subrede e qual a quantidade máxima de hosts disponíveis para utilização dentre outras informações. Esta ferramenta *open-source* foi desenvolvida em C# por Micu Radu e está disponível no site CodeProject. A Figura 13 mostra a interface de usuário do *NetCalculator*.

Figura 13 - Interface de Usuário do NetCalculator.



Fonte: RADU, 2005.

3.5.4 NETWORKMINER

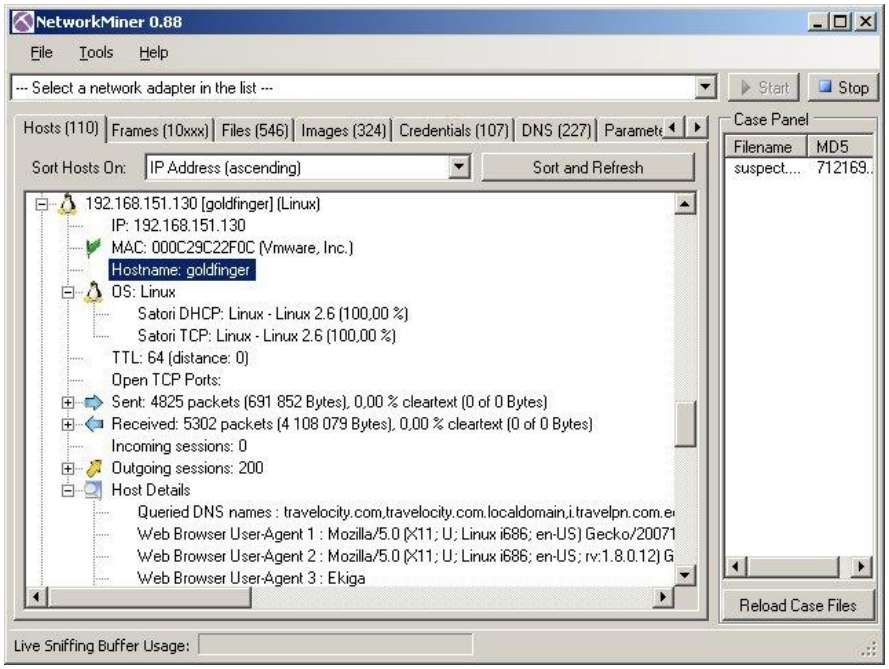
A ferramenta *NetworkMiner* é um *software* NFAT (*Network Forensic Analysis Tool* – Ferramenta de Análise Forense de Rede) desenvolvido originalmente para o ambiente *Windows*. Ela é produzida e distribuída pela Netresec e sua primeira versão foi lançada em 2007. Atualmente é possível executar o *NetworkMiner* em ambientes Linux, Mac OS X e FreeBSD. Esta ferramenta é utilizada por profissionais e organizações em todo o mundo e é citada, por exemplo, pela revista PCWorld (KIRK, 2014) como uma das ferramentas utilizadas para identificar falhas de privacidade de usuário em aplicativos de redes sociais.

O NetwokMiner recebe constantes atualizações e funciona como um *sniffer* de rede passivo com o objetivo de capturar o tráfego e identificar sistemas operacionais, sessões, *hostnames*, portas abertas e outras informações sobre a rede. Este *software* tem compatibilidade com redes IPV4 e IPV6 e também possui uma versão paga que contém inúmeras outras funcionalidades (NETRESEC, 2017).

Esta ferramenta pode auxiliar o profissional que realiza o NTA (*Network Traffic Analysis* – Análise de Tráfego de Rede), pois oferece uma interface intuitiva e simplificada para a visualização dos dados. O código fonte que foi escrito em C# (versão 6.0) e está

disponível para *download* no site da Netresec. A Figura 14 ilustra a interface gráfica do *NetworkMiner*.

Figura 14 - Interface Gráfica do NetworkMiner.



Fonte: SECTOOLS, 2011.

3.5.5 JNETMAP

O *jNetMap* é uma ferramenta *open-source* desenvolvida utilizando a linguagem Java. Este *software* possibilita a criação de uma representação gráfica da topologia de uma rede de computadores. Além disso, ele possui uma funcionalidade que verifica periodicamente se um dispositivo continua conectado. O *jNetMap* ainda é capaz de realizar o descobrimento de rede, o escaneamento de portas em uso e possui uma estrutura para adicionar novas funcionalidades através de *plug-ins*. Alguns como o *SSH Connection* (*Secure Shell Connection*[7]), *Remote Desktop RDP* (*Remote Desktop Plug-in* – Plug-in de Área de Trabalho Remota), *Wake On Lan*[8] e vários outros *plug-ins* de notificações já estão integrados originalmente à ferramenta. Os plug-ins de notificação permitem o envio de *e-mail*, a reprodução de um aviso sonoro ou uma notificação de área de trabalho nativa do sistema operacional quando um dispositivo

[7] O protocolo Secure Shell (SSH) é um protocolo de rede criptografado para realizar operações de forma segura sobre uma rede insegura. Mais informações disponíveis em: <http://www.ssh.com/>.

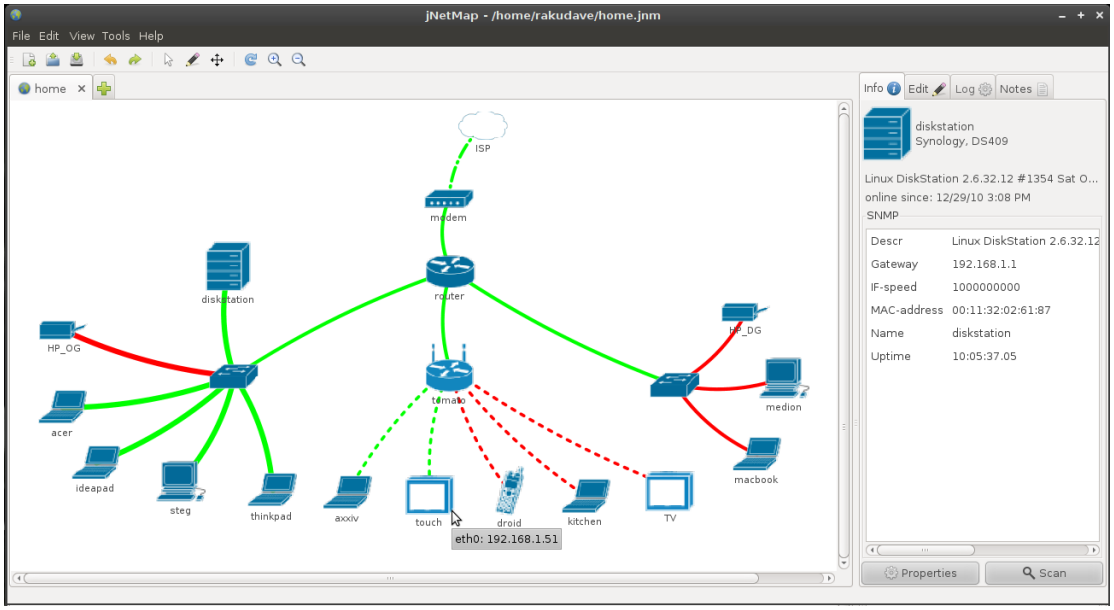
[8] Wake-on-LAN permite que um computador seja ligado por uma mensagem de rede. Para correto funcionamento a interface de rede deve estar em *stand-by*. Mais informações estão disponíveis em: <http://www.dei.isep.ipp.pt/~andre/documentos/wol.html >.

previamente mapeado fica *online* ou *offline*. A lista completa de *plug-ins* está disponível no site oficial da ferramenta[9].

Para verificar se um dispositivo está conectado, o *jNetMap* utiliza diferentes formas do comando *ping* (RAKUDAVE, 2011). Automaticamente a ferramenta vai identificar qual a melhor forma e executará o comando de acordo com a rede e o sistema operacional utilizado. O *Java Ping* é o modo padrão, porém este comando pode ter alguns problemas em versões mais antigas do Java que estejam rodando no ambiente Windows. Para garantir o funcionamento multiplataforma, o desenvolvedor do *jNetMap* implementou o *System Ping*. Esta implementação executa o comando de *ping* nativo do sistema operacional de acordo com a plataforma em que o *software* está sendo executado. Apesar de ser um método um pouco mais lento, ele garante a funcionalidade em múltiplas plataformas (RAKUDAVE, 2011).

GRAÇAS (2013), renomado profissional de TI e professor de Volta Redonda (Rio de Janeiro), indica que o *jNetMap* não é uma ferramenta para auditoria de rede, mas sim uma ferramenta para um centro de controle simplificado e intuitivo. O *jNetMap* continua em desenvolvimento pelo programador alemão Rakudave e em maio deste ano teve sua última atualização. A Figura 17 ilustra a interface principal do *jNetMap* e algumas de suas funcionalidades.

Figura 15 - Interface Gráfica do jNetMap.



Fonte: RAKUDAVE, 2011.

[9] A lista completa de *plug-ins* pode ser encontrada em: < http://www.rakudave.ch/jnetmap/?file=plugins >.

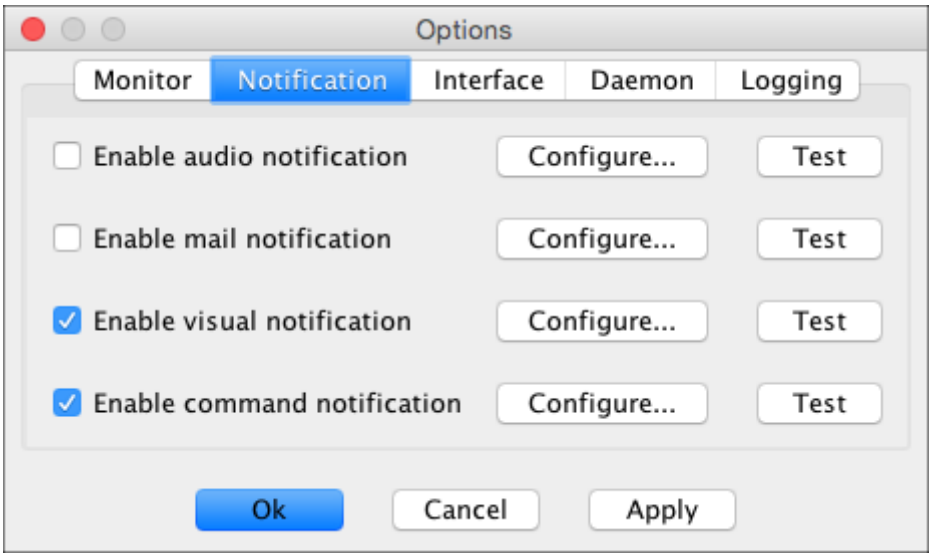
3.5.6 IP MONITOR

O *IP Monitor* é uma ferramenta *open-source* desenvolvida em linguagem Java. O *IP Monitor* é um simples *software* que permite monitorar o endereço de IP público utilizado e notificar o administrador de rede em casos de mudanças. É possível configurar diversos tipos de notificações tais como: *e-mail*, notificação visual, notificação por áudio e outras (ZANETTI, 2017).

O *IP Monitor* pode ser executado como um serviço do Windows ou como um *daemon*[10] do Linux. Ele foi desenvolvido pelo programador argentino Gabriel Zanetti e continua recebendo atualizações.

A Figura 16 ilustra a interface de notificações do IP Monitor.

Figura 16 - Configuração de Notificações do IP Monitor.



Fonte: ZANETTI, 2017.

3.6 CONSIDERAÇÕES FINAIS

Ao finalizar a pesquisa exploratória com a exposição de ferramentas pagas e gratuitas, conclui-se que a quantidade de estudos sobre o conceito BYOD é relativamente pequena

[10] Daemon é a definição semelhante aos Serviços do Windows porém no ambiente Linux. Maiores informações podem ser encontradas em: <https://www.vivaolinux.com.br/artigo/Entendendo-um-pouco-sobre-os-daemons>.

justamente por se tratar de um conceito recente. Esta foi a maior dificuldade durante o cumprimento do cronograma de atividades.

As ferramentas foram selecionadas de acordo com a classificação e tipologia citada na Seção 3.3 deste capítulo. Diante das quatro classificações possíveis, o objetivo foi selecionar pelo menos uma ferramenta que se enquadrasse em cada opção.

De forma geral, todas estas seis ferramentas *open-source* que foram listadas nesta subseção são compatíveis com ambientes de redes BYOD. Realizando a combinação entre o modelo BYOD e ferramentas de código aberto, as corporações podem diminuir os custos de TI drasticamente. Caso alguma corporação ainda necessite de algum complemento, existem os softwares que são gratuitos para uso, mas não são *open-source*.

A listagem destas ferramentas foi concebida com o intuito de projetar um protótipo que auxilie o profissional responsável pela segurança e administração da rede em uma corporação. Através de inúmeras buscas, estas ferramentas resultantes parecem estar de acordo com a proposta inicial do trabalho.

4. ESTRUTURAÇÃO DO PROTÓTIPO

Este capítulo detalha o protótipo que foi desenvolvido para integrar ferramentas BYOD disponíveis no mercado. Para realizar esta tarefa, os códigos fontes foram analisados. Na etapa seguinte, algumas técnicas de programação foram utilizadas para o desenvolvimento da solução proposta. Este protótipo de *software* servirá de auxílio para o cumprimento das políticas de administração e segurança de redes que possam estar sendo adotadas pela corporação.

O desenvolvimento de um protótipo capaz de integrar as ferramentas do tópico ferramentas *open-source* (Seção 3.5) e sua definição estrutural foi exposto nas próximas seções. Para garantir a compatibilidade entre diferentes linguagens, a comunicação entre processos foi feita utilizando *sockets* (Seção 4.2). As definições tecnológicas (Seção 4.3) e as definições do protótipo (Seção 4.4) foram apresentadas na sequência.

4.1 FERRAMENTAS UTILIZADAS

Para realizar o desenvolvimento do protótipo, algumas ferramentas são necessárias. A linguagem de programação escolhida é a linguagem Java. O ambiente de desenvolvimento integrado (IDE – *Integrated Developtment Environment*) que será utilizado é o Eclipse. A principal razão para escolha destas plataformas é o conhecimento e experiência prévia que o autor possui. Além disso, por se tratar também de uma ferramenta *open-source*, a comunidade costuma manter o Eclipse atualizado com certa frequência.

A Sun Microsystems[11] foi a responsável pela criação da linguagem Java no ano de 1991. Originalmente, a ideia era construir uma linguagem de programação para pequenos dispositivos, como por exemplo, controles de TV e celulares da época. Desde o primeiro momento ela foi planejada para ser compatível com os mais diferentes tipos de equipamentos, fazendo com que eles se comunicassem entre si. Atualmente a linguagem não é utilizada em eletrodomésticos, como planejada, mas se tornou umas das linguagens de programação mais utilizadas no mundo (MENGUE, 2002).

[11] A Oracle anunciou a compra da Sun Mircrosystems, e consequentemente da linguagem Java também, em 2009 por cerca de US$ 7 bilhões. Mais informações estão disponíveis em: <http://g1.globo.com/Noticias/Tecnologia/0,,MUL1091457-6174,00-ORACLE+ANUNCIA+COMPRA+DA+SUN+POR+MAIS+DE+US+BILHOES.html>.

A vantagem da linguagem Java é executar em uma máquina virtual (JVM – *Java Virtual Machine*). No momento em que o código é compilado ele é convertido em um código intermediário chamado *bytecode*. O *bytecode* funciona como instruções de máquina para a JVM. Uma vez compilado, o *bytecode* pode ser interpretado em qualquer dispositivo que possua uma JVM (ORACLE, 2017).

A plataforma Java é dividida em dois componentes principais: a JVM e a Java API (*Application Programming Interface* – Interface de Programação de Aplicações). A API é um grupo de bibliotecas com classes e interfaces, estruturas de manipulação de dados e arquivos entre outras funcionalidades (MENGUE, 2002).

Para o desenvolvimento em Java da aplicação proposta neste trabalho, o uso do Eclipse e de suas ferramentas será essencial. A versão escolhida é a Eclipse Neon (4.6) que foi lançada no dia 22 de junho de 2017. Uma ferramenta interessante do Eclipse é a WindowBuilder. Esta ferramenta possibilita a construção de interfaces de usuário utilizando as principais APIs gráficas disponíveis, como por exemplo, a Java Swing. O *plug-in* WindowBuilder possibilita a criação de telas de maneira ágil e produtiva sem gastar muito tempo escrevendo código (ECLIPSE, 2017).

4.2 NETWORK SOCKET

Como citado anteriormente, o *NetworkMiner* (Seção 3.5.4) foi desenvolvido em linguagem C#. Para realizar a integração desta e outras linguagens com a linguagem Java, foram utilizados *network sockets* (Soquetes de Rede).

O *network socket* é o ponto final para envio e recebimento de dados por um único nodo em uma rede de computadores. Seu objetivo é permitir que dois processos troquem informações entre si. Esta comunicação é conhecida como “Cliente-Servidor” e possibilita que programas desenvolvidos em diferentes linguagens troquem informações sem grandes dificuldades.

Um *socket* é composto por uma combinação entre um endereço IP e o número de uma porta. O número da porta é definido de acordo com o protocolo de transporte (TCP ou UDP) que será utilizado (Richard Stevens, Fenner e Rudoff, 2003).

Os sistemas operacionais possuem, de forma nativa, uma API *Socket* que pode ser acessada com poucas linhas de comandos em inúmeras linguagens de programação. Isso facilita o desenvolvimento de *softwares* com estrutura cliente-servidor.

4.3 DEFINIÇÕES DE TECNOLOGIA

Para a execução deste trabalho, foi necessário estabelecer um ambiente digital favorável. A arquitetura necessária foi composta por um ambiente de desenvolvimento e um ambiente de testes.

O ambiente de desenvolvimento do protótipo é a plataforma Windows de 64 bits, através do uso do Eclipse. Os requisitos mínimos para o funcionamento do Java no Windows são: processador Pentium 2 266 MHz, 128MB de memória RAM e 181MB de espaço livre em disco (ORACLE, 2016). Os requisitos mínimos para o Eclipse Neon são: 300MB de espaço livre em disco, 1GB de memória RAM e Java 8 ou superior (ECLIPSE, 2017).

O ambiente de testes foi composto por uma cópia do sistema operacional Windows de 64 bits. Esta cópia foi executada em uma máquina virtual através do VMware. No ambiente de testes foi possível executar o protótipo de integração e analisar os resultados obtidos após execução das ferramentas propostas.

Para a execução das ferramentas *open-source* citadas na Seção 3.5 foi necessário que o sistema oferecesse:

- Java versão 8 ou superior
- GCC (GNU *Compiler Collection*)
- Microsoft .NET versão 2.0 ou superior

Como algumas ferramentas foram desenvolvidas em outras linguagens orientadas a objeto, foi necessário a refatoração de trechos de código utilizando a IDE apropriada.

4.4 DEFINIÇÕES DO PROTÓTIPO

A especificação do *software* é responsável por estabelecer quais funções são necessárias e quais as restrições sobre a operação e desenvolvimento do sistema (SOMMERVILLE, 2011). Nesta etapa, foram coletados os requisitos necessários para a modelagem do sistema. A organização dos requisitos em grupos de diferentes níveis possibilitou a divisão do *software* em camadas.

Os requisitos funcionais devem descrever explicitamente quais são as finalidades, funcionalidades e serviços do sistema. Eles são capazes de documentar como o sistema deve

atuar em determinadas entradas de dados e/ou como o sistema deve se comportar em situações específicas, além de descrever o que o sistema não deve fazer.

Já os requisitos não funcionais são os que definem as propriedades e restrições do *software*. Eles podem definir características do sistema todo ou só de partes dele. É importante salientar que estes requisitos podem ser mais críticos que os requisitos funcionais. Por exemplo, se um *software* não satisfizer um nível "X" de segurança e desempenho então ele é inútil (FIGUEIREDO, 2016).

O protótipo de integração de ferramentas de administração e segurança BYOD visa atender algumas necessidades básicas, tais como: oferecer uma boa solução BYOD, ser intuitivo, funcionar corretamente, manter a integridade dos dados e evitar possíveis falhas. Além disso, a manutenção do mesmo deve ser realizada de forma fácil.

Os requisitos funcionais deste protótipo são:

- Permitir a captura de pacotes de rede: responsável por oferecer um *sniffer* de rede que permita a análise dos dados trafegados.
- Permitir o gerenciamento de senhas: responsável por oferecer uma ferramenta que armazena usuários e senhas em apenas uma base de dados criptografada.
- Informar o nível de segurança de uma senha: responsável por oferecer uma ferramenta capaz de classificar as senhas de acesso entre senhas fracas e senhas fortes.
- Agilizar os cálculos de redes IPV4: responsável por oferecer uma ferramenta capaz de realizar cálculos de endereços de rede de forma ágil.
- Permitir a varredura de dispositivos conectados na rede: responsável por oferecer uma ferramenta com capacidade de realizar a descoberta de rede e mapear graficamente os dispositivos.
- Permitir a verificação de endereço IP: responsável por oferecer uma ferramenta para monitoramento do endereço de IP público.

Os requisitos não funcionais são:

- Manter as licenças de *software*: garantir que as licenças *open-source* de cada ferramenta integrada serão respeitadas.
- Manter a compatibilidade com a plataforma: garantir que o protótipo funcione corretamente no ambiente Windows.
- Manter a integridade dos dados: garantir que os dados e arquivos pessoais continuem seguros e operando corretamente.

- GUI (*Graphical User Interface* – Interface Gráfica de Usuário) de fácil utilização: oferecer uma interface de usuário *desktop* que seja acessível e fácil de utilizar.

- Documentação de Auxílio: oferecer um serviço de ajuda bom o suficiente para evitar a necessidade de treinamento de usuário.

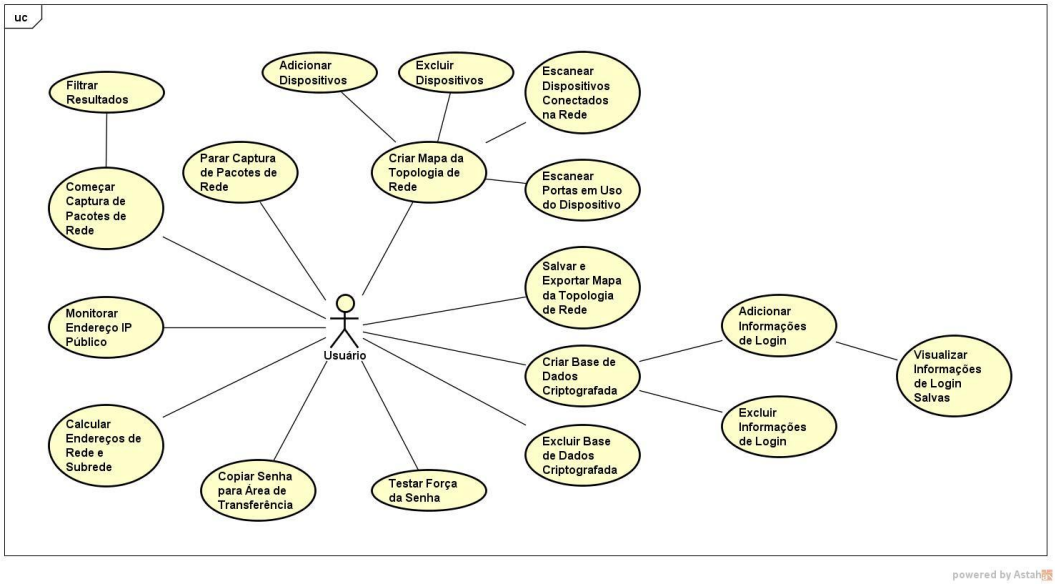
O público alvo deste projeto de integração foram os profissionais de TI de corporações de pequeno porte que permitam o conceito BYOD no ambiente de trabalho, além de usuários domésticos avançados que queiram usufruir das facilidades propostas pela suíte de gerenciamento e segurança.

4.4.1 DIAGRAMAS DE CASOS DE USO

Para ilustrar as necessidades citadas anteriormente, os diagramas de casos de uso foram utilizados. Desta forma é possível compreender facilmente quais situações ocorreram durante o uso da ferramenta. Os casos de uso tendem a facilitar o entendimento de quais tarefas precisam ser feitas pelos desenvolvedores de *software*.

A Figura 17 ilustra o diagrama de casos de uso que o protótipo apresenta para integração de ferramentas de administração e segurança BYOD.

Figura 17 - Casos de Uso do Protótipo.



Fonte: Autoria própria.

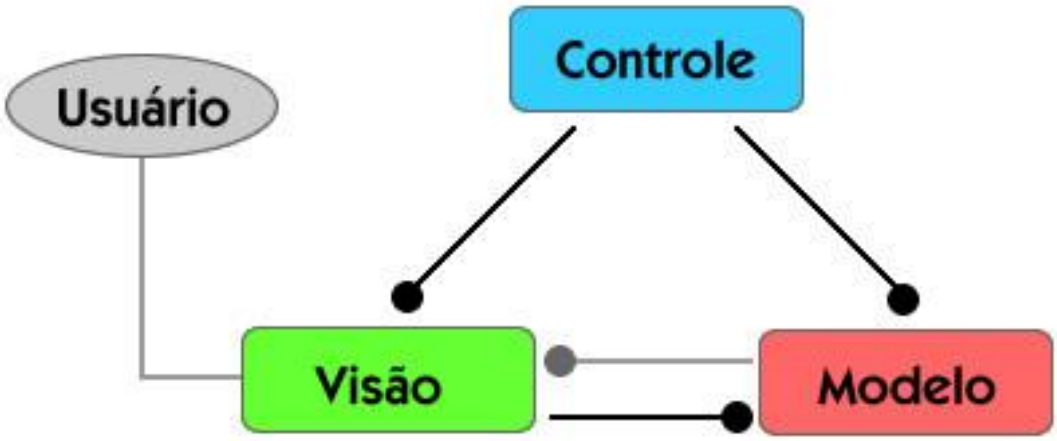
As descrições dos casos de uso ilustrados na Figura 17 estão no anexo A.

4.4.2 CAMADAS DE SOFTWARE

O protótipo foi formado por três camadas de software, seguindo a ideia do modelo MVC (*Model View Controller*). O modelo MVC é estruturado através de três camadas que podem interagir entre si. A camada modelo (*model*) é responsável pelas regras de negócio, persistência e dados da aplicação. A camada visão (*view*) é capaz de mostrar os dados ao usuário e de receber novas instruções. A camada controlador (*controller*) é responsável por receber instruções da visão, conferir se estão corretas, e enviar para o modelo, além de receber comandos do modelo e enviar para a representação realizada na visão (GULZAR, 2002).

Apesar de o MVC ser fortemente utilizado para aplicações *web*, seu modelo de camadas serviu como base para a definição da arquitetura do protótipo *desktop* proposta. As demais definições propostas pelo MVC não foram aproveitadas. A Figura 18 ilustra o modelo de camadas MVC.

Figura 18 - Modelo de Camadas MVC.



Fonte: CELESTINO, 2014.

4.5 CONSIDERAÇÕES FINAIS

A funcionalidade multiplataforma proposta pela linguagem Java, tende a facilitar a exportação deste protótipo para outras plataformas e sistemas operacionais. Todas as

ferramentas selecionadas podem funcionar em diferentes ambientes e sistemas operacionais, proporcionando uma maior abrangência e facilitando a adoção dos *softwares* por corporações que utilizem o BYOD.

Com a utilização do modelo MVC, as interfaces do protótipo podem ser facilmente inseridas ou modificadas de acordo com a necessidade do usuário. Estas alterações não alteram o funcionamento nativo das ferramentas, pois elas ficam alocadas na camada Modelo.

O funcionamento do protótipo no ambiente Windows é garantido através da utilização conjunta do IDE Eclipse e das demais definições de tecnologia e seus respectivos requisitos.

5. DESENVOLVIMENTO DO PROTÓTIPO

A integração das ferramentas *open-source*, citadas na seção 3.5, foi realizada com o objetivo de oferecer praticidade, agilidade e comodidade ao profissional de TI. Supondo um ambiente empresarial de pequeno porte, o administrador de rede precisa manter diversos serviços e *softwares* sendo executados simultaneamente para realizar o monitoramento e gerenciamento da rede. Além disso, ele pode enfrentar problemas de compatibilidade entre os *softwares*, impossibilitando a troca de informações rapidamente. Com a integração das ferramentas, o profissional tem acesso a uma única suíte com vários *softwares* que atendem suas demandas e que podem trocar dados entre si de maneira ágil.

O desenvolvimento do protótipo de integração de ferramentas para administração e segurança BYOD foi realizado de acordo com os modelos descritos no decorrer do Capítulo 4 (Estruturação do Protótipo). A interface gráfica do protótipo foi desenvolvida garantindo o funcionamento em diferentes resoluções. Alguns itens citados no Capítulo 4 foram alterados no momento em que o protótipo foi desenvolvido. O processo de desenvolvimento e todas as alterações do projeto estão relatados nas próximas seções.

5.1 CONFIGURAÇÕES DO AMBIENTE DE DESENVOLVIMENTO

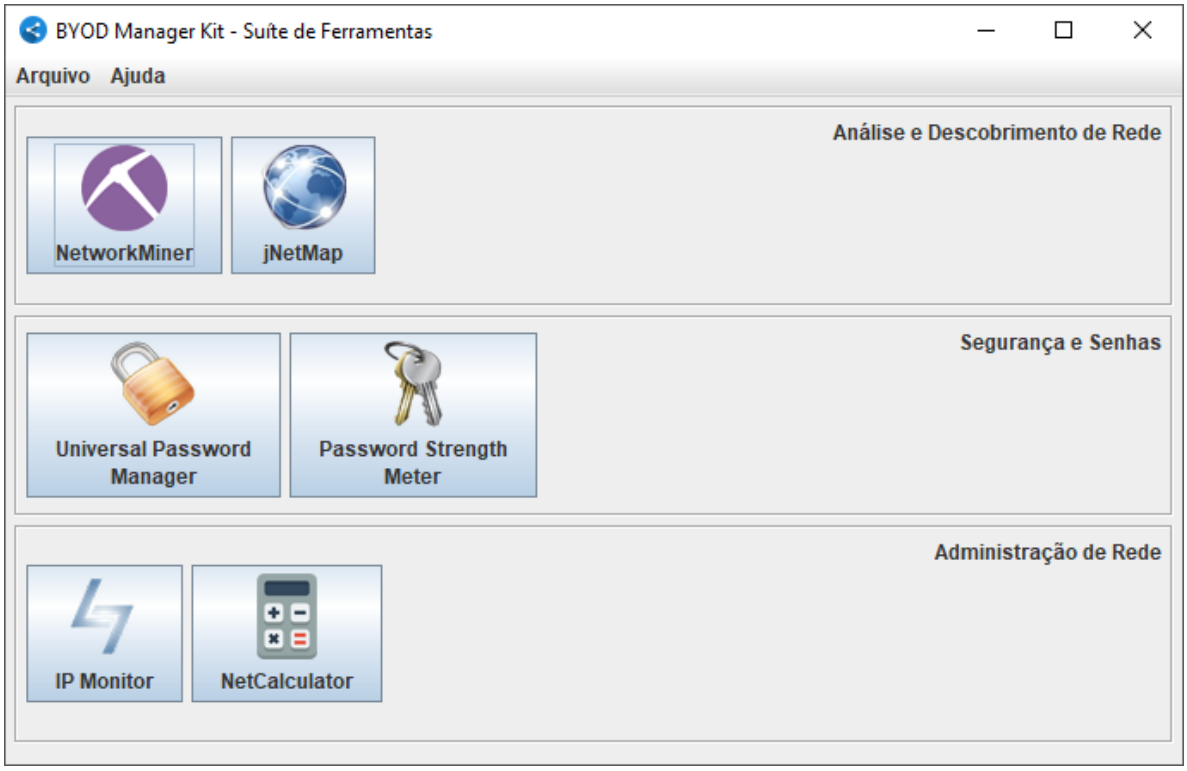
Para o pleno desenvolvimento do projeto os seguintes requisitos foram instalados e configurados:

- Máquina virtual utilizando software VMware: Sistema Operacional Windows 7 SP2;
- JDK (*Java SE Development Kit*) 9;
- Microsoft .*NET Framework* 4.5;
- Linguagem de Programação: Java e C# 6.0;
- WindowBuilder *plug-in;*
- Eclipse Neon IDE;
- Visual Studio 2017 *Community* IDE.

5.2 ESTRUTURA DE FUNCIONAMENTO

O protótipo de integração, nomeado de *BYOD Manager Kit*, visa à criação de uma suíte de ferramentas para a administração e gerenciamento de redes. Através desta suíte é possível acessar as ferramentas *open-source* escolhidas na seção 3.5 através dos ícones ou via menu “Arquivo”. Além disso, toda a documentação oficial das ferramentas está disponível no menu “Ajuda”. Também no menu “Ajuda”, é possível acessar a opção “Sobre” que disponibiliza algumas informações [12] sobre o projeto, as licenças *open-source* utilizadas, créditos aos autores das ferramentas e um manual de usuário, em formato PDF, para o protótipo. A Figura 19 ilustra a interface principal da suíte. A interface foi desenvolvida para oferecer o correto funcionamento em diversas resoluções de tela. Todas as telas foram desenvolvidas utilizando o *plug-in* WindowBuilder do Eclipse IDE.

Figura 19- Interface Principal do Protótipo.



Fonte: Próprio Autor, 2017.

[12] Breves explicações sobre as funcionalidades oferecidas no protótipo e porque o projeto está sendo desenvolvido. Incluindo informações creditando a Universidade de Caxias do Sul e os professores orientadores e avaliadores do trabalho de conclusão de curso.

Durante o desenvolvimento do protótipo, as seis ferramentas foram classificadas em três áreas de atuação: a área de “Análise e Descobrimento de Rede”, a área de “Segurança e Senhas” e a área de “Administração de Rede”.

A área “Análise e Descobrimento de Rede” é formada pelas ferramentas *NetworkMiner* e *jNetMap*. Essa combinação foi feita porque o *sniffer* de rede *NetworkMiner* pode auxiliar diretamente na visualização de tráfego da rede. Já o *jNetMap*, além de poder testar as portas em utilização, possui uma ferramenta para descoberta de rede. Através da integração, estas ferramentas podem trocar algumas informações entre si, incorporando novas funcionalidades. Este processo é detalhado na seção 5.3.1 (Integração entre o *NetworkMiner* e o *jNetMap*) .

A área “Segurança e Senhas” é formada pelas ferramentas *Universal Password Manager* e *Password Strength Meter*. A união foi feita porque ambas as ferramentas trabalham diretamente com credenciais de usuário e senhas. O administrador de rede poderá armazenar, em uma base de dados criptografada, suas senhas de acesso a servidores, roteadores e *switches* gerenciáveis, suas credenciais de rede e outras senhas em geral. O processo de integração entre as duas ferramentas é detalhado na seção 5.3.2.

A área “Administração de Rede” é formada pelas ferramentas *IP Monitor* e *NetCalculator*. Esta junção foi realizada, pois as ferramentas selecionadas possuem finalidades administrativas. O profissional de TI poderá utilizar estas ferramentas para realizar a configuração da sua estrutura de rede. Os detalhes da integração destes dois *softwares* estão disponíveis nas próximas seções.

Em paralelo com a interface do usuário, a suíte executa um processo servidor (*Server Socket* [13]) na porta 7000 do *localhost*. Este processo foi desenvolvido em Java, assim como todo o resto da suíte principal. Ele é responsável por garantir a comunicação entre processos que ocorrem quando mais de uma ferramenta da suíte é utilizada ao mesmo tempo. Desta forma, foi possível garantir a troca de informações entre o Java e o C#, por exemplo. Algumas funcionalidades que foram adicionadas às ferramentas *open-source* utilizam este canal para a troca de mensagens. Todo o detalhamento de implementação destas funcionalidades também é descrito nas próximas seções.

O Anexo C é composto pelo manual de usuário da ferramenta *BYOD Manager Kit*. Neste manual é possível acompanhar passo a passo todas as funcionalidades do *software*.

[13] “ServerSocket” é o nome da classe Java utilizada para a criação do processo servidor. A documentação oficial da classe e outras informações estão disponíveis em: <https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html>.

É importante ressaltar que a arquitetura e modelagem de software utilizados nas ferramentas *open-source* foram mantidas de acordo com o código dos autores. As modificações e inclusões de novos códigos buscaram respeitar o modelo MVC (seção 4.4.2).

5.2.1 FLUXO DE COMUNICAÇÃO ENTRE PROCESSOS

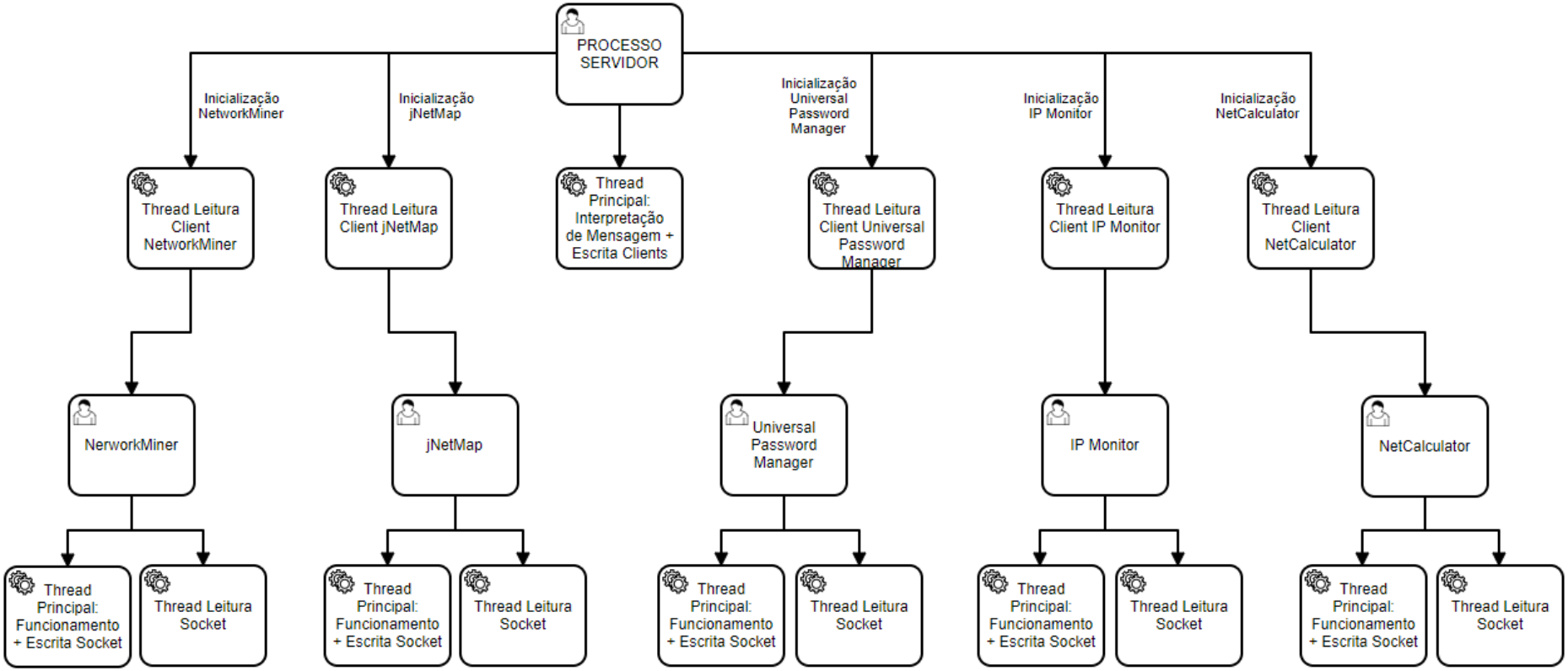
Ao criar um sistema de comunicação entre as ferramentas *open-source* foi necessário gerenciar o fluxo de comunicação destes processos. O processo servidor é inicializado juntamente com a interface principal da suíte. Os clientes são inicializados e conectados no momento em que o usuário abrir as ferramentas.

A Figura 20 ilustra o fluxo de comunicação, as *threads* criadas e suas respectivas funções.

O fluxo de comunicação entre os processos ocorre através da utilização de várias *threads* e *sockets*. Os nodos com engrenagens representam as *threads*. Os nodos com usuários representam as ferramentas integradas. O evento de “Inicialização” ilustrado na Figura 20 corresponde ao clique nos ícones das ferramentas. No momento em que uma ferramenta é inicializada, o servidor cria uma *thread* somente para a leitura do socket. O processo cliente segue a mesma lógica, criando uma *thread* somente para leitura. As escritas no *socket* são feitas pela *thread* principal das ferramentas. Esse modelo foi adotado por ser difícil prever quando e/ou em qual ordem o usuário enviará informações de uma ferramenta para outra.

A ferramenta *Password Strength Meter* não utiliza a comunicação por *sockets*, pois ela é apenas uma biblioteca que foi adicionada ao projeto principal e ao projeto *Universal Password Manager*.

Figura 20 - Fluxograma de Comunicação



Fonte: Próprio Autor, 2017.

5.2.2 DIAGRAMA DE CLASSES

Os diagramas de classes foram divididos de acordo com as três camadas do MVC. Mais detalhes sobre as camadas MVC estão disponíveis nas próximas subseções.

Um diagrama foi criado para a camada visão, contendo as principais classes e métodos utilizados no desenvolvimento da interface de usuário. Outro diagrama contendo as informações mais importantes foi criado para a camada controle. Para a camada modelo foi criado um diagrama que contém todos os detalhes das modificações e implementações realizadas nas ferramentas *open-source*. As figuras dos diagramas e breves explicações das funções que foram desenvolvidas estão presentes no anexo B.

5.2.3 CAMADA MODELO

Na camada modelo do protótipo ficam os arquivos fonte das ferramentas *open-source* que foram selecionadas. As ferramentas que são desenvolvidas em Java foram salvas em formato JAR[14] (Java Archive – Arquivo Java) e adicionadas a esta camada. Com exceção da ferramenta *Password Strength Meter* que teve os arquivos incluídos em formato Java. As ferramentas que são desenvolvidas em C# tiveram seus respectivos arquivos executáveis (formato EXE[15]) e bibliotecas (formato DLL[16]) incluídas na camada modelo.

Em nenhuma ferramenta o código fonte desta camada foi alterado. A arquitetura criada pelos desenvolvedores oficiais foi mantida.

5.2.4 CAMADA VISÃO

Na camada visão foram criadas as interfaces gráficas do protótipo. Foram desenvolvidas diferentes classes Java, responsáveis por painéis de mensagens, caixas de diálogos e janelas principais. Todas essas interfaces foram desenvolvidas utilizando o *plug-in WindowBuilder*. A Figura 21 mostra a interface gráfica que o *WindowBuilder* proporciona no Eclipse IDE e consequentemente sua facilidade e agilidade no desenvolvimento deste tipo de

[14] O formato JAR é um formato utilizado para agrupar vários arquivos de classes (formato .java) e recursos associados (imagens, ícones, pdfs, etc) em um só arquivo. Ele pode ser utilizado como um arquivo executável para distribuição de programas (semelhante ao formato .exe do Windows). Mais informações em: <https://docs.oracle.com/javase/tutorial/deployment/jar/index.html>.
[15] Extensão de arquivo que denota um arquivo executável no ambiente Windows.
[16] Dynamic-Link Library é o formato utilizado no ambiente Windows para retratar as bibliotecas compartilhadas entre processos.

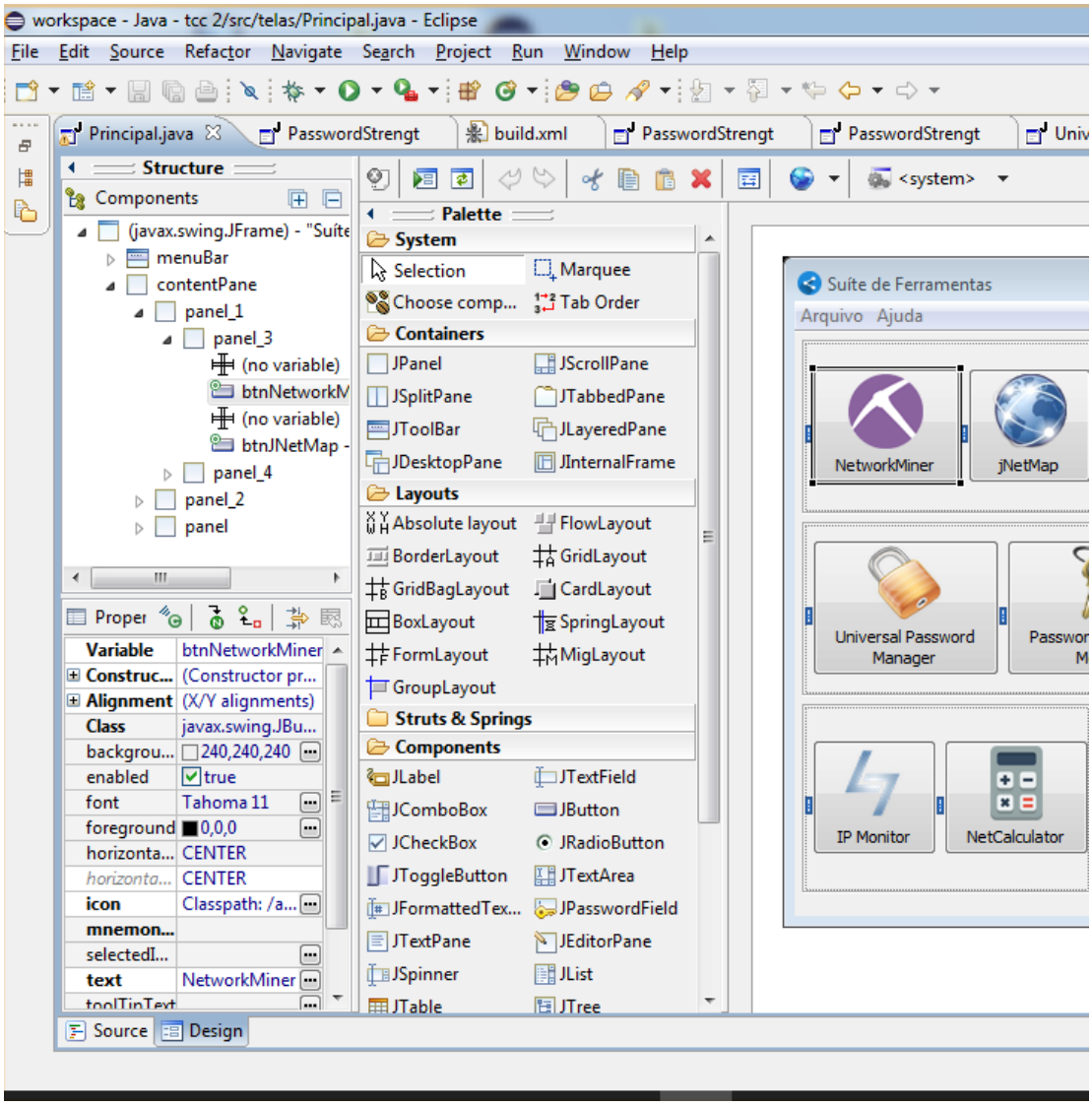
objeto. Através de uma visualização gráfica, o WindowBuilder gera o código fonte de acordo com os objetos e suas propriedades, que são adicionados na tela. Por fim, é possível destacar as classes Principal.java, Sobre.java e a PasswordStrengthMeterGUI.java.

A Classe Principal.java é a classe responsável pela função *main* e por criar a janela principal da interface gráfica. Na função *main* é criada uma instância da classe *SocketServer* (Camada Controle) para possibilitar a integração de todas as ferramentas. Esta classe é responsável pelo gerenciamento dos eventos de usuário e por mostrar possíveis erros. Por exemplo, quando o usuário clica no botão "*NetworkMiner*", a interface interpreta o evento de clique realizando a abertura da ferramenta (caso a classe PermiteAbrirFrame da Camada Controle retorne positivamente) e em seguida comunica a classe *SocketServer* que realiza a conexão com o cliente.

Classe Sobre é a classe responsável por um diálogo que contém algumas informações sobre o protótipo. Um breve texto foi desenvolvido explicando porque a ferramenta foi desenvolvida e creditando a Universidade de Caxias do Sul e os professores envolvidos no trabalho. O diálogo também contém uma breve explicação sobre as licenças *open-source* que foram utilizadas no protótipo e nas ferramentas que foram integradas. Além disso, os desenvolvedores das ferramentas são creditados.

Classe PasswordStrengthMeterGUI.java é uma interface gráfica que foi desenvolvida especialmente para a biblioteca Password Strength Meter. Ela possibilita que o usuário teste suas senhas de acordo com diferentes padrões, informa um número aproximado de iterações para que a senha seja quebrada e ainda possibilita copiar a senha rapidamente para a área de transferência através de um botão.

Figura 21 - Interface Gráfica do WindowBuilder; Editando a classe Principal.java.



Fonte: Próprio Autor, 2017.

5.2.5 CAMADA CONTROLE

A camada controle foi estruturada através de três classes: Documentacao.java, PermiteAbrirFrame.java e SocketServer.java. A camada controle é responsável pelo gerenciamento da quantidade de ferramentas que estão abertas, pela chamada de inicialização das ferramentas, além do gerenciamento de comunicação entre processos clientes e o processo servidor. Por fim, a documentação original das ferramentas também é tratada nesta camada.

A classe Documentação.java é responsável por verificar se o sistema utilizado possui um software para leitura de PDFs e então permitir a leitura da documentação do NetworkMiner e do jNetMap. As demais ferramentas possuem a documentação online, então a classe verifica a existência de um navegador *web* no sistema.

A classe PermiteAbrirFrame.java é a classe que controla se uma ferramenta está ou não aberta, impedindo o usuário de abrir mais de uma vez a mesma ferramenta. Além disso, essa é a classe que faz a chamada para a inicialização das ferramentas (armazenadas na camada modelo).

A classe SocketServer.java é a classe que cria o servidor *socket* e permite a conexão dos clientes. Cada cliente é conectado respeitando um sistema de identificação "IDs" previamente definido no código. É temporariamente impossível abrir uma segunda ferramenta enquanto a primeira ainda não estabeleceu a conexão com o servidor. Esse pequeno bloqueio, que ocorre durante uma fração de segundos, garante o correto funcionamento da comunicação entre processos no protótipo. Esta classe possui a função "InterpretaMensagem(String msg, int id)" que recebe a mensagem que deverá ser interpretada e o "id" que representa qual ferramenta enviou esta mensagem. A mensagem pode significar informações que devem ser enviadas para outra ferramenta ou até comunicar o fechamento de uma ferramenta.

5.3 ALTERAÇÕES NO CÓDIGO FONTE DAS FERRAMENTAS

Para que a comunicação entre as ferramentas e o processo servidor ocorresse, foram realizadas algumas alterações no código-fonte destas ferramentas. Para as ferramentas desenvolvidas em Java e C#, foram criadas duas classes "SocketClient", uma para cada linguagem. Como as informações são enviadas ao processo servidor e recebidas pelo processo cliente em formato String[17], o ambiente Java e o ambiente C# conseguem trocar dados sem a necessidade de um tratamento específico.

A classe "SocketClient" foi adicionada aos projetos *open-source* respeitando o modelo MVC. Como a maioria das ferramentas selecionadas possui esta arquitetura de *software*, a classe "SocketClient" foi adicionada à camada controle destes projetos. Para seguir o padrão de desenvolvimento, a camada controle foi criada nos projetos que não utilizavam o modelo MVC.

Além disso, todas as demais modificações e novas funcionalidades, como a criação de um filtro de resultados no *NetworkMiner*, a fila de importação de dispositivos no *jNetMap*, entre outras, foram feitas nas suas respectivas camadas de *software*. Estas funcionalidades estão explícitas nas próximas seções.

[17] String é uma estrutura de dados composta por uma cadeia de caracteres. Maiores informações sobre o tratamento de Strings em Java está disponível em: <https://docs.oracle.com/javase/7/docs/api/java/lang/String.html>. A documentação de Strings em C# está disponível em: <https://msdn.microsoft.com/pt-br/library/system.string(v=vs.110).aspx>.

Foram necessárias alterações na camada visão do projeto *NetworkMiner*, pois um novo menu foi adicionado à tela principal da ferramenta. O mesmo ocorreu nas ferramentas *jNetMap* e *Universal Password Manager*.

A ferramenta *NetCalculator* não necessitou da inclusão de novos elementos na interface gráfica, mas algumas mudanças, como por exemplo o tratamento de eventos, foram realizadas na camada visão.

A ferramenta *Password Strength Meter* não possui uma interface gráfica. A camada visão do protótipo de integração foi utilizada para criar uma interface que pudesse facilitar o uso da biblioteca pelo usuário final.

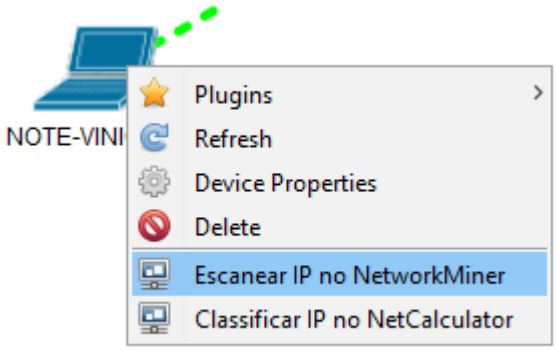
A ferramenta *IP Monitor* sofreu alterações somente na camada controle com a adição da classe "SocketClient".

5.3.1 INTEGRAÇÃO ENTRE O NETWORKMINER E O JNETMAP

A integração entre o *NetworkMiner* e o *jNetMap* foi desenvolvida com o intuito de possibilitar a troca de informações entre as duas ferramentas. Através dela é possível enviar dados do *NetworkMiner* para o *jNetMap* e/ou enviar dados do *jNetMap* para o *NetworkMiner*. Na ferramenta *NetworkMiner*, foi desenvolvido um filtro de resultados e um processo de escaneamento automático. Na ferramenta *jNetMap*, foi desenvolvida uma fila de importação de dispositivos.

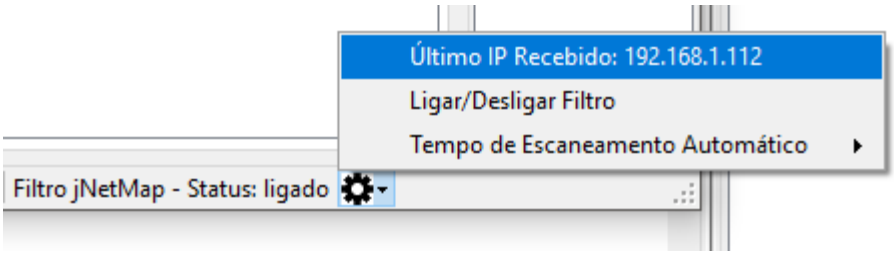
O filtro de resultados do *NetworkMiner* é ativado automaticamente quando o usuário envia um endereço IP a partir do *jNetMap*. Para realizar este envio, o usuário precisa clicar com o botão direito sobre um dispositivo e escolher a opção "Escanear IP no *NetworkMiner*". A Figura 22 ilustra a opção "Escanear IP no *NetworkMiner*" que foi adicionada ao menu de dispositivo na ferramenta *jNetMap*. A Figura 23 ilustra o recebimento do endereço IP no *NetworkMiner* e indica que o filtro está ligado. Neste momento, o usuário deve clicar no botão "*Start*" no *NetworkMiner* para começar a captura de pacotes relacionados ao endereço IP. Para finalizar a captura, é necessário clicar no botão "*Stop*". A Figura 24 ilustra o campo de seleção de adaptador de rede e os botões "*Start*" e "*Stop*". Para desativar o filtro é necessário clicar no botão "Ligar/Desligar Filtro", localizado no menu "Filtro *jNetMap*".

Figura 22 - Detalhe da opção "Escanear IP no NetworkMiner" adicionada ao jNetMap.



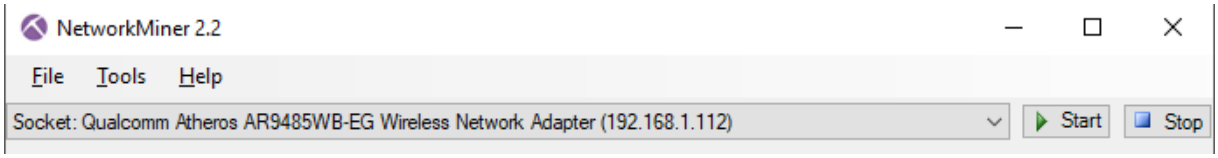
Fonte: Próprio Autor, 2017.

Figura 23 - Detalhe do menu "Filtro jNetMap" adicionado ao NetworkMiner.



Fonte: Próprio Autor, 2017.

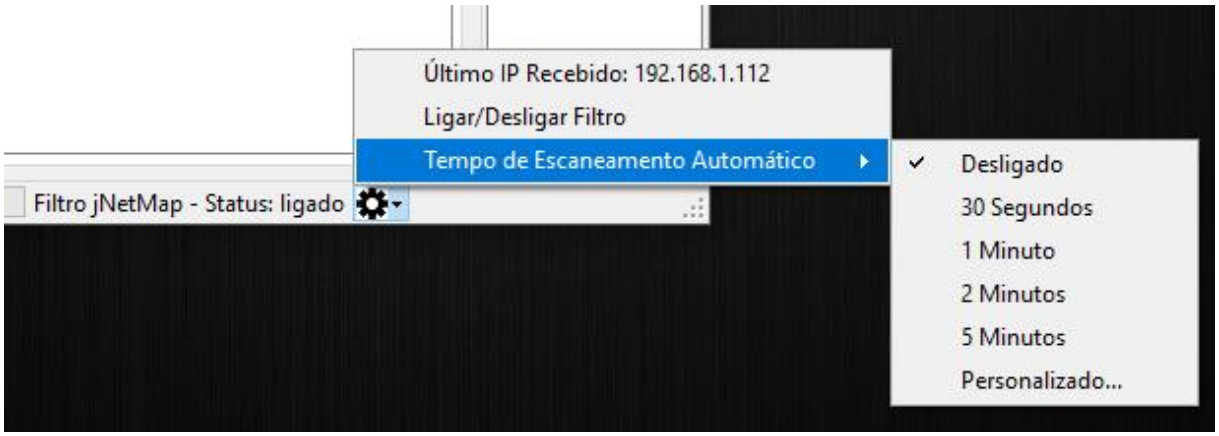
Figura 24 - Detalhe dos botões "Start" e 'Stop".



Fonte: Próprio Autor, 2017.

O escaneamento automático foi desenvolvido no *NetworkMiner* com o intuito de remover a necessidade do usuário clicar nos botões "*Start*" e "*Stop*". Para que o escaneamento automático ocorra quando o *NetworkMiner* recebe um endereço IP, o usuário precisa configurar alguns itens previamente. Um adaptador de rede precisa estar selecionado, além disso, o usuário deverá informar por quanto tempo a captura de pacotes será executada. O usuário pode escolher entre um intervalo de tempo pré-definido ou por um intervalo personalizado. A Figura 25 ilustra o menu "Tempo de Escaneamento Automático".

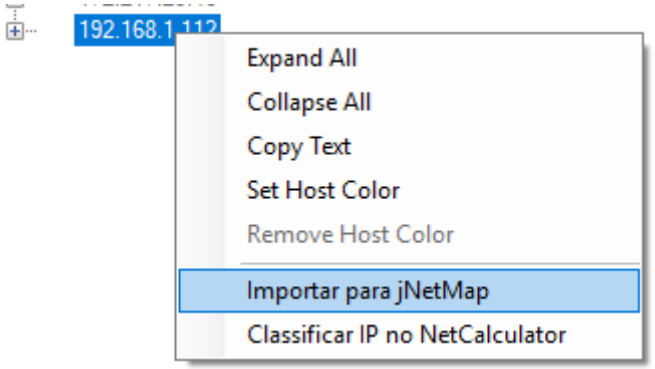
Figura 25 - Detalhe do menu "Tempo de Escaneamento Automático".



Fonte: Próprio Autor, 2017.

A fila de importação de *hosts* foi criada no *jNetMap* com o objetivo de facilitar a adição de novos dispositivos ao mapa de rede. Esta fila segue o padrão FIFO (*First in, First Out* – Primeiro a Entrar, Primeiro a Sair). Para alimentar a fila, é necessário que o usuário clique com o botão direito do mouse sobre um *host* no *NetworkMiner* e clique na opção "Importar para *jNetMap*". A Figura 26 ilustra a opção "Importar para *jNetMap*". No momento em que o usuário clicar nesta opção, o *NetworkMiner* irá reunir informações como nome do computador, endereço IP, fabricante e outras e enviará estes dados para o *jNetMap*.

Figura 26 - Detalhe da opção "Importar para jNetMap".

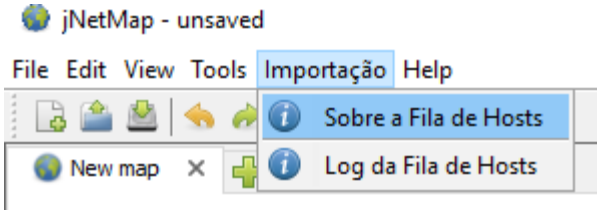


Fonte: Próprio Autor, 2017.

Quando os dados são recebidos pelo *jNetMap*, eles são armazenados na fila de importação. O usuário pode consultar o *log* de recebimento e o status atual da fila pelo menu "Importação". A Figura 27 ilustra o menu "Importação" no *jNetMap*. Caso exista um dispositivo na fila, é possível adicioná-lo a qualquer mapa de rede. Para realizar a importação, o usuário deve clicar com o botão direito do mouse em qualquer área em branco do mapa

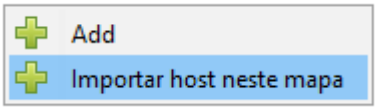
desejado e selecionar a opção “Importar *host* neste mapa”. A Figura 28 ilustra a opção “Importar *host* neste mapa”. Então, o usuário deve completar o cadastro de informações e clicar em “OK” para adicionar o dispositivo ao mapa. A Figura 29 ilustra o cadastro presente no menu de importação. Além disso, é possível notar na Figura 29 quais campos foram automaticamente preenchidos com informações obtidas pelo *NetworkMiner*.

Figura 27 - Detalhe do menu "Importação" adicionado ao jNetMap.



Fonte: Próprio Autor, 2017.

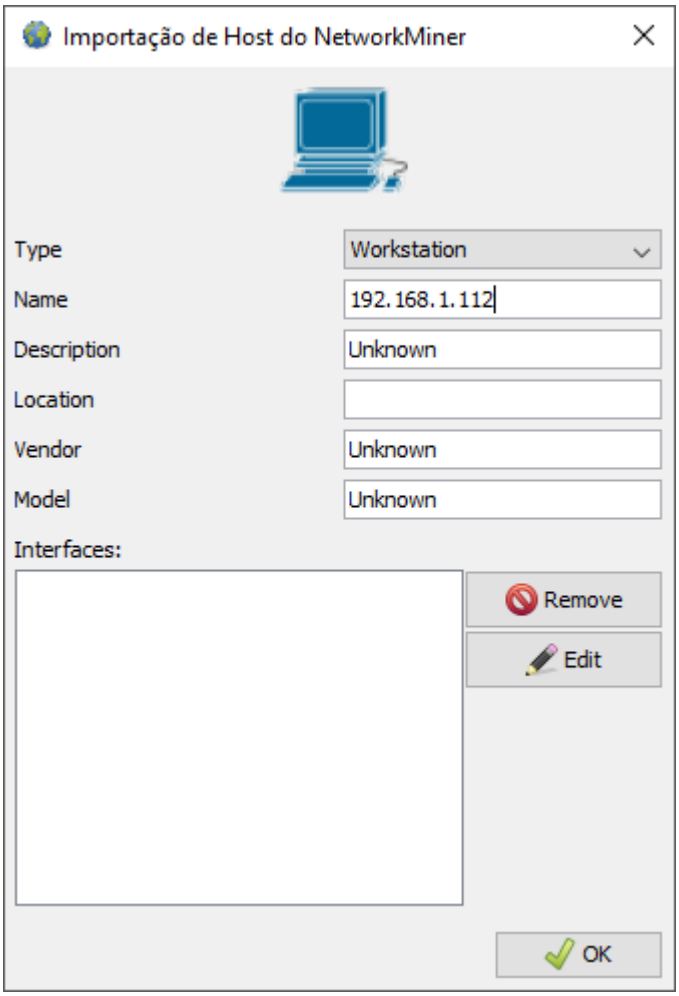
Figura 28 - Detalhe da opção "Importar host neste mapa".



Fonte: Próprio Autor, 2017.

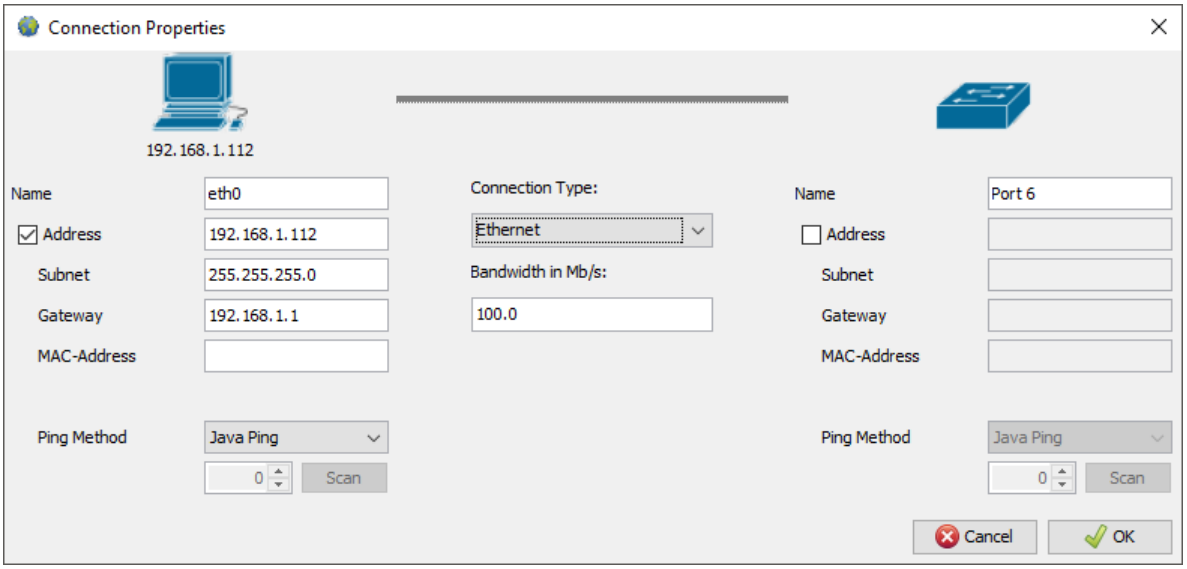
Após a adição do dispositivo ao mapa, o usuário pode realizar a conexão dele com a estrutura de rede. Para estabelecer esta conexão o usuário deve utilizar a ferramenta “Lápis”, clicar sobre o dispositivo e arrastar até o ponto de conexão desejado. A Figura 30 ilustra a conexão do dispositivo recém importado ao *switch* de rede.

Figura 29 - Detalhe do cadastro presente no menu de importação de host.



Fonte: Próprio Autor, 2017.

Figura 30 - Detalhe da conexão do host ao switch de rede.



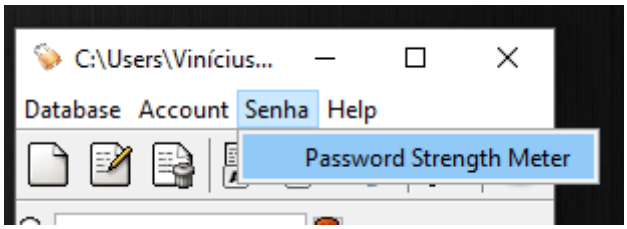
Fonte: Próprio Autor, 2017.

5.3.2 INTEGRAÇÃO ENTRE UNIVERSAL PASSWORD MANAGER E PASSWORD STRENGTH METER

A integração entre o *Universal Password Manager* e o *Password Strength Meter* foi realizada com o objetivo de criar uma funcionalidade que permitisse testar as senhas armazenadas. Um menu chamado “Senha” foi adicionado à interface principal do *Universal Password Manager*. A Figura 31 ilustra o menu “Senha”. O usuário pode utilizar tanto o menu “Senha” quanto o ícone “*Password Strength Meter*” na interface principal do protótipo para abrir a ferramenta de teste de senhas.

Uma interface gráfica foi desenvolvida para a biblioteca *Password Strength Meter* e ela pode ser observada na Figura 33.

Figura 31 - Detalhe do menu "Senha" adicionado ao Universal Password Manager.



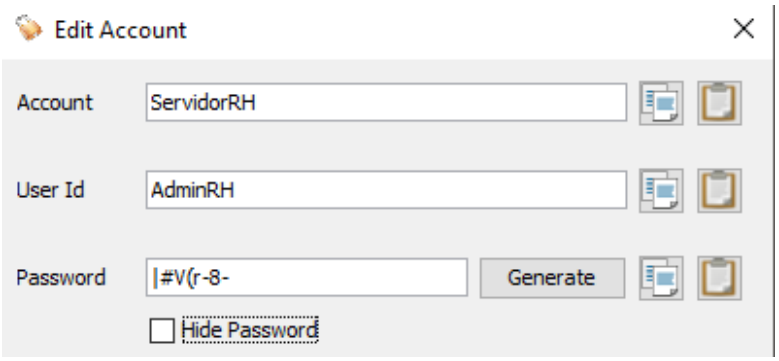
Fonte: Próprio Autor, 2017.

O usuário pode testar suas próprias senhas ou as senhas geradas automaticamente pelo *Universal Password Manager*. Além disso, ele pode escolher o tamanho das senhas geradas e várias outras opções através do menu “*Options*”, ilustrado na Figura 34.

Para gerar automaticamente uma senha, é necessário clicar no botão “*Generate*” (Figura 32). O botão “*Generate*” está disponível tanto na adição de um novo cadastro, quanto na visualização das credenciais já salvas.

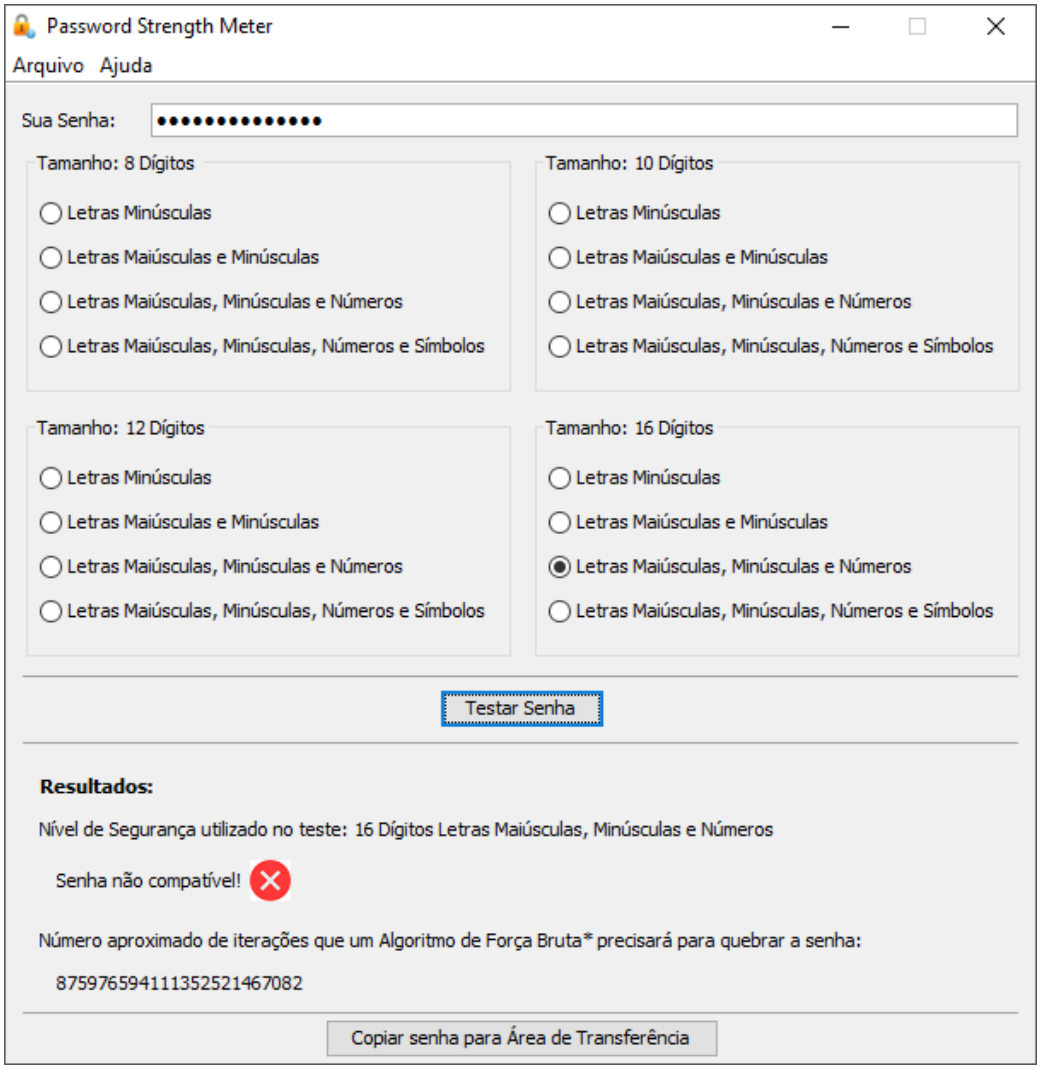
Para testar uma senha, é necessário informá-la no campo “Sua Senha” e após selecionar algum dos requisitos que a senha deve atender, como ilustrado na Figura 32. Feito isso, o usuário deve clicar no botão “Testar Senha”. Os resultados serão exibidos logo abaixo, de forma detalhada. O usuário ainda pode copiar a senha de maneira rápida através do botão “Copiar senha para Área de Transferência”.

Figura 32 - Detalhe do botão "Generate".



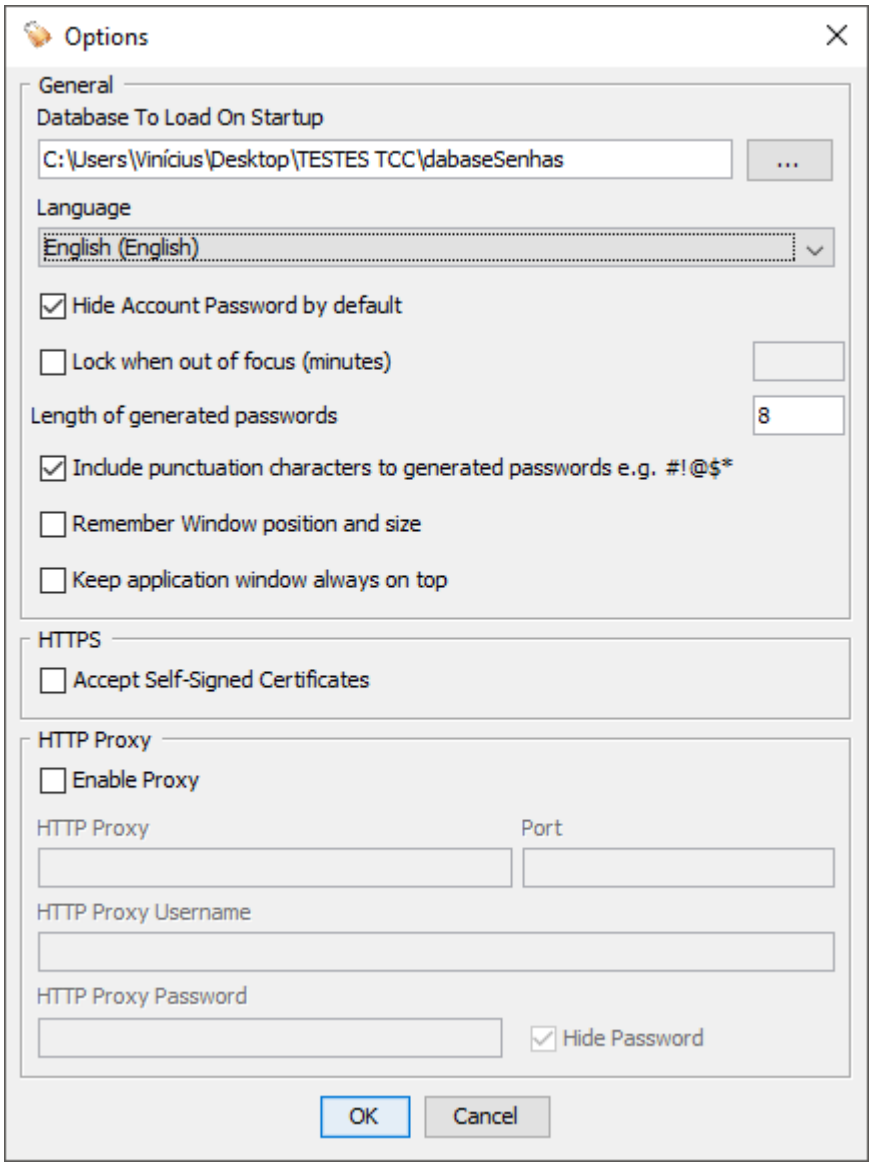
Fonte: Próprio Autor, 2017.

Figura 33 - Interface gráfica da biblioteca Password Strength Meter.



Fonte: Próprio Autor, 2017.

Figura 34 - Detalhe do menu "Options" da ferramenta Universal Password Manager.

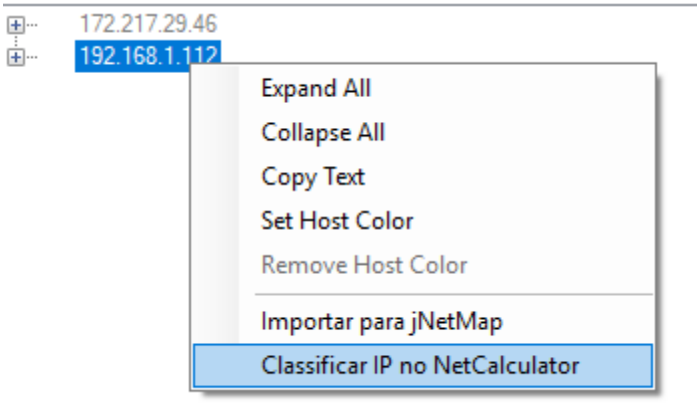


Fonte: Próprio Autor, 2017.

5.3.3 INTEGRAÇÃO ENTRE NETCALCULATOR, NETWORKMINER E JNETMAP

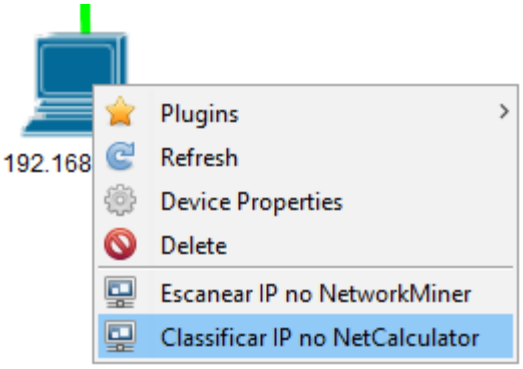
A integração entre o *NetCalculator*, o *NetworkMiner* e o *jNetMap* possibilitou o recebimento de informações da rede BYOD em tempo real pelo *NetCalculator*. Foi adicionada ao *NetworkMiner* e ao *jNetMap* uma opção chamada "Classificar IP no *NetCalculator*". Esta opção pode ser acessada, nas duas ferramentas, clicando com o botão direto do mouse sobre um dispositivo ou *host*. As Figuras 35 e 36 ilustram a opção "Classificar IP no *NetCalculator*" nas duas ferramentas.

Figura 35 - Detalhe da opção "Classificar IP no NetCalculator" adicionada ao NetworkMiner.



Fonte: Próprio Autor, 2017.

Figura 36 - Detalhe da opção "Classificar IP no NetCalculator" adicionada ao jNetMap.



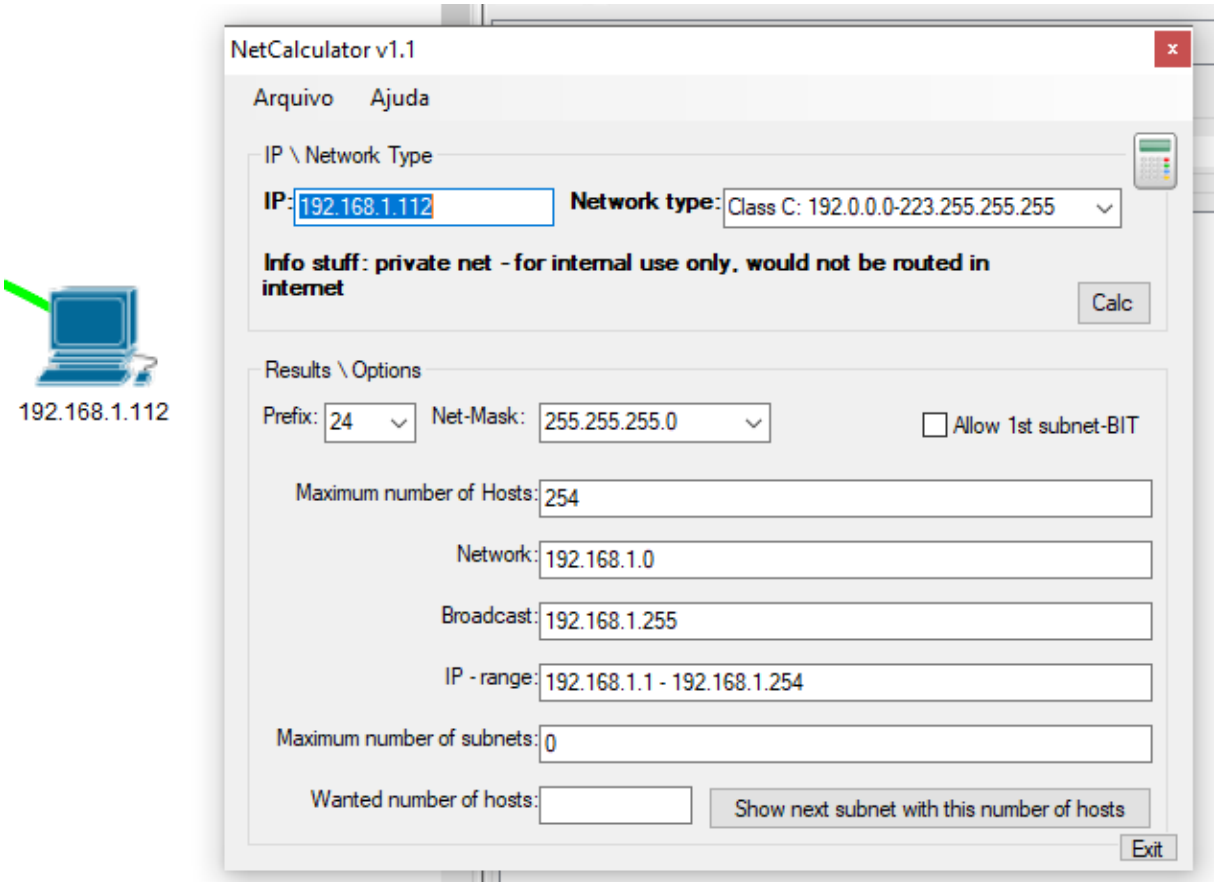
Fonte: Próprio Autor, 2017.

Ao receber um endereço IP, automaticamente o *NetCalculator* vai realizar os cálculos de rede e informar os resultados na sua interface gráfica. Entre os resultados é possível destacar: o número máximo de *hosts* da rede, o endereço *broadcast* e o endereço de rede, além da faixa de endereços IPs. A Figura 37 ilustra um endereço IP que foi enviado do *jNetMap* para o *NetCalculator*. Por limitações de ferramenta, o *NetworkMiner* não consegue descobrir qual a máscara de rede utilizada em um endereço IP. Portanto, a integração foi desenvolvida de modo que apenas o endereço IP é enviado ao *NetCalculator*. O *NetCalculator* define a máscara de rede através do padrão imposto pela classe que o endereço IP possui. Porém, o usuário pode alterar livremente a máscara de rede caso seja necessário.

Algumas alterações foram realizadas na interface do *NetCalculator* para tornar a ferramenta mais acessível. A Figura 38 ilustra uma mensagem de atenção mostrando ao

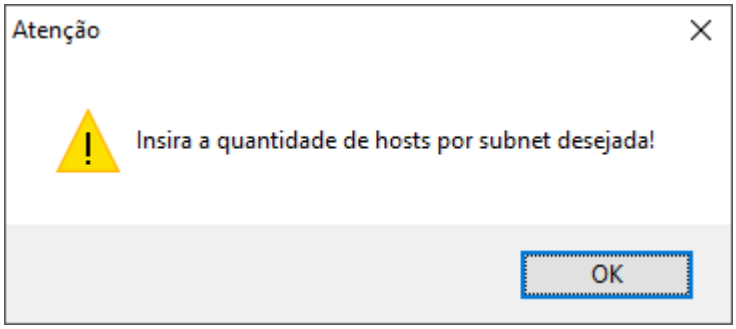
usuário que ele deve adicionar o número de hosts por subrede antes de clicar no botão "*Show next subnet with this number of hosts*".

Figura 37 - Interface principal do NetCalculator mostrando os resultados após cálculo baseado em endereço IP enviado pelo jNetMap.



Fonte: Próprio Autor, 2017.

Figura 38 - Detalhe da mensagem de atenção que foi adicionada ao NetCalculator.



Fonte: Próprio Autor, 2017.

6. TESTES E RESULTADOS

Durante o desenvolvimento do protótipo, alguns pequenos testes foram realizados de acordo com o progresso da integração entre as ferramentas. Estes testes foram realizados no ambiente de testes definido na seção 4.3. Quando o desenvolvimento do protótipo foi finalizado, a ferramenta foi testada inúmeras vezes nesta máquina virtual, durante longos períodos sem interrupções. Após os testes neste ambiente, a ferramenta foi instalada em um ambiente real. O ambiente escolhido foi o Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, como citado na seção 1.3. A documentação e resultados destes testes estão descritos a seguir.

Os testes foram divididos em três etapas. A etapa inicial testou apenas as funcionalidades originais das ferramentas. A segunda etapa testou as funcionalidades que foram desenvolvidas e adicionadas no protótipo de integração. A última etapa simulou ambientes reais de acordo com as quatro grandes categorias de vulnerabilidades BYOD (seção 2.2). Além disso, os testes foram realizados em redes sem fio (Wi-Fi) e redes cabeadas (*Ethernet*).

6.1 TESTES DE FUNCIONALIDADES DAS FERRAMENTAS

Nesta etapa, as ferramentas tiveram suas funcionalidades originais testadas. O *BYOD Manager Kit* foi executado com “Permissões de Administrador” e os resultados são descritos abaixo.

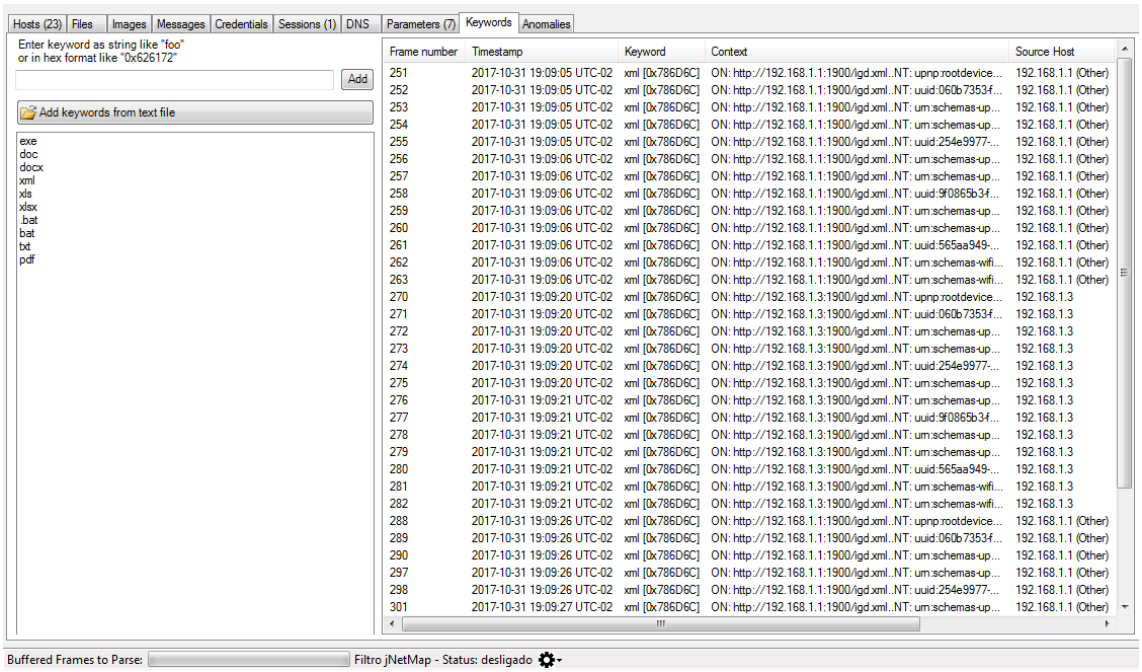
O primeiro teste realizado foi: executar o *NetworkMiner* para escanear a rede durante quinze minutos utilizando as seguintes *keywords* para interceptar dados em específico: “exe”, “bat”, “pdf”, “xml”, “doc”, “docx”, “xls”, “xlsx” e “txt”. Os resultados obtidos foram:

- Rede Wi-Fi - Máquina Virtual, Windows 7: A ferramenta funcionou normalmente durante o período de escaneamento. A Figura 39 ilustra os arquivos “xml” que foram interceptados no momento em que o navegador Google Chrome foi inicializado.

- Rede Wi-Fi - Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, Windows 10 (*notebook* próprio): A ferramenta funcionou normalmente durante o período de escaneamento. A Figura 40 ilustra os *hosts* que foram interceptados durante o escaneamento. A Figura 41 ilustra os detalhes do endereço de *Broadcast* 10.20.63.255.

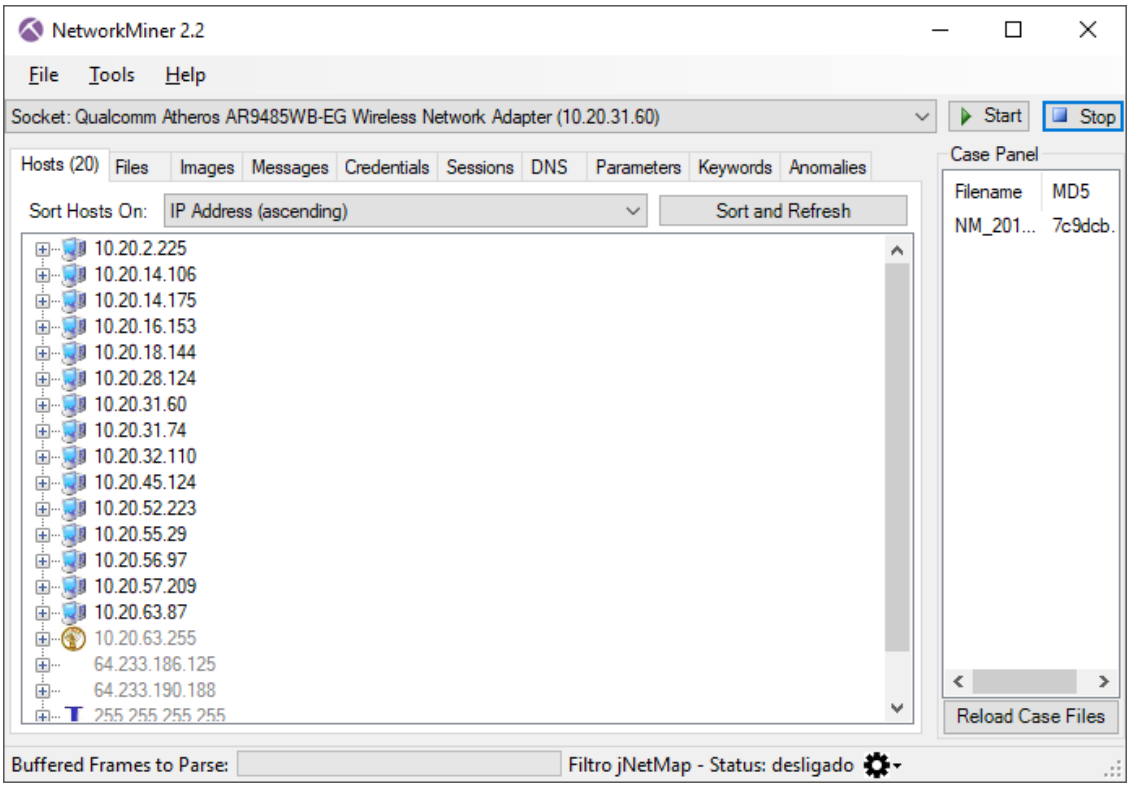
- Rede *Ethernet* - Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, Windows 7 (*desktop* da UCS): A ferramenta apresentou instabilidades e teve sua execução bloqueada pelo antivírus da Universidade de Caxias do Sul. As Figuras 42 e 43 ilustram os problemas apresentados durante a execução do *NetworkMiner*.

Figura 39 - Arquivos XML capturados pelo NetworkMiner.



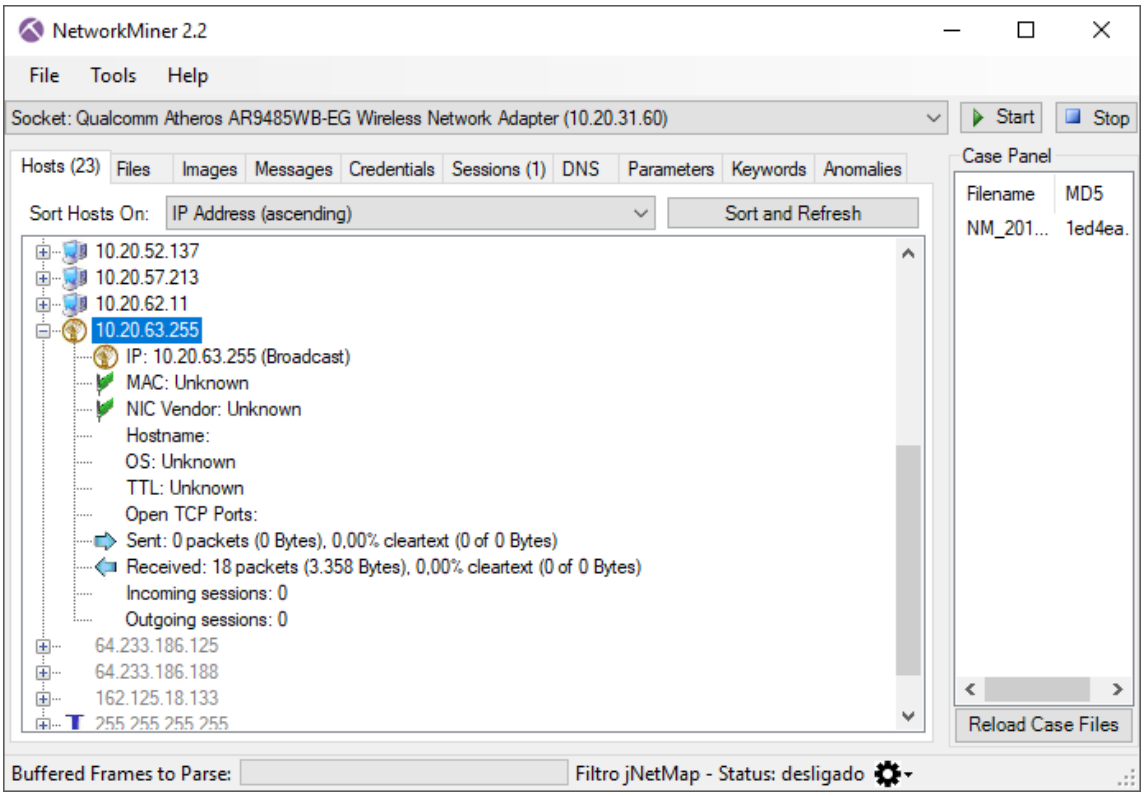
Fonte: Próprio Autor, 2017.

Figura 40 - Hosts escaneados no NetworkMiner.



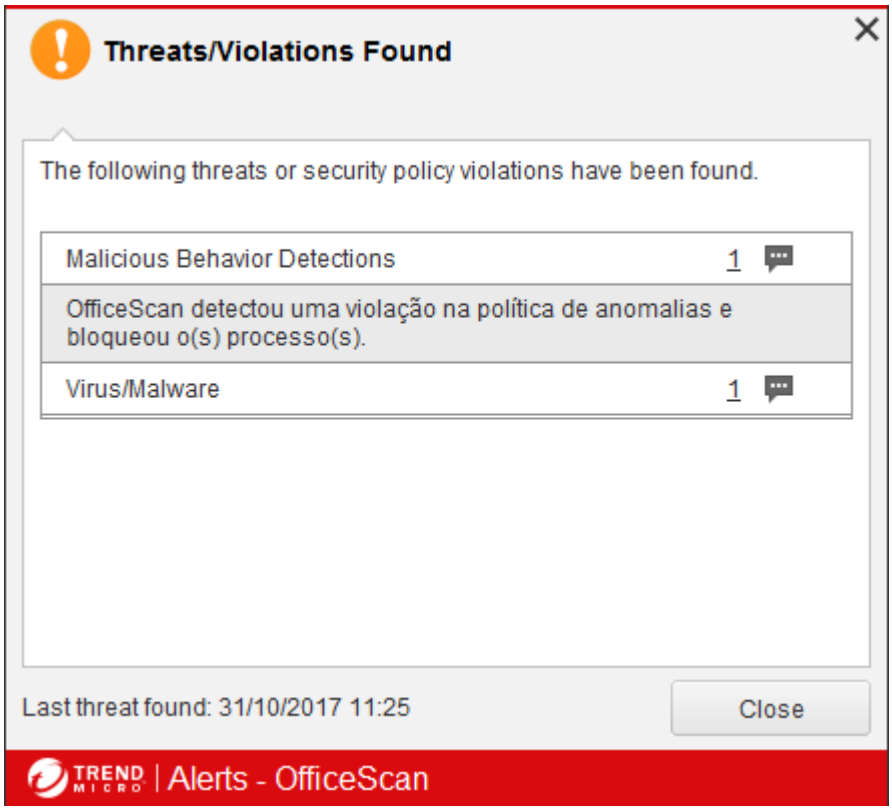
Fonte: Própria Autoria, 2017.

Figura 41 - Informações complementares do host.



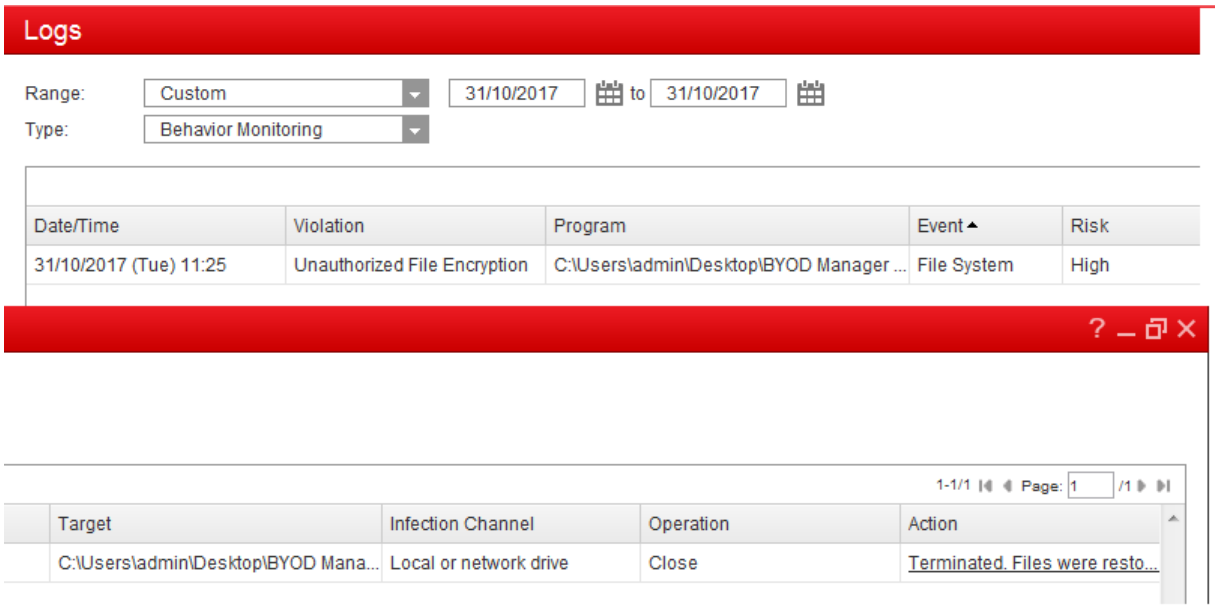
Fonte: Próprio Autor, 2017.

Figura 42 - Problemas com o antivírus da UCS.



Fonte: Próprio Autor, 2017.

Figura 43 - Detalhamento dos problemas com o antivírus da UCS.



Fonte: Próprio Autor, 2017.

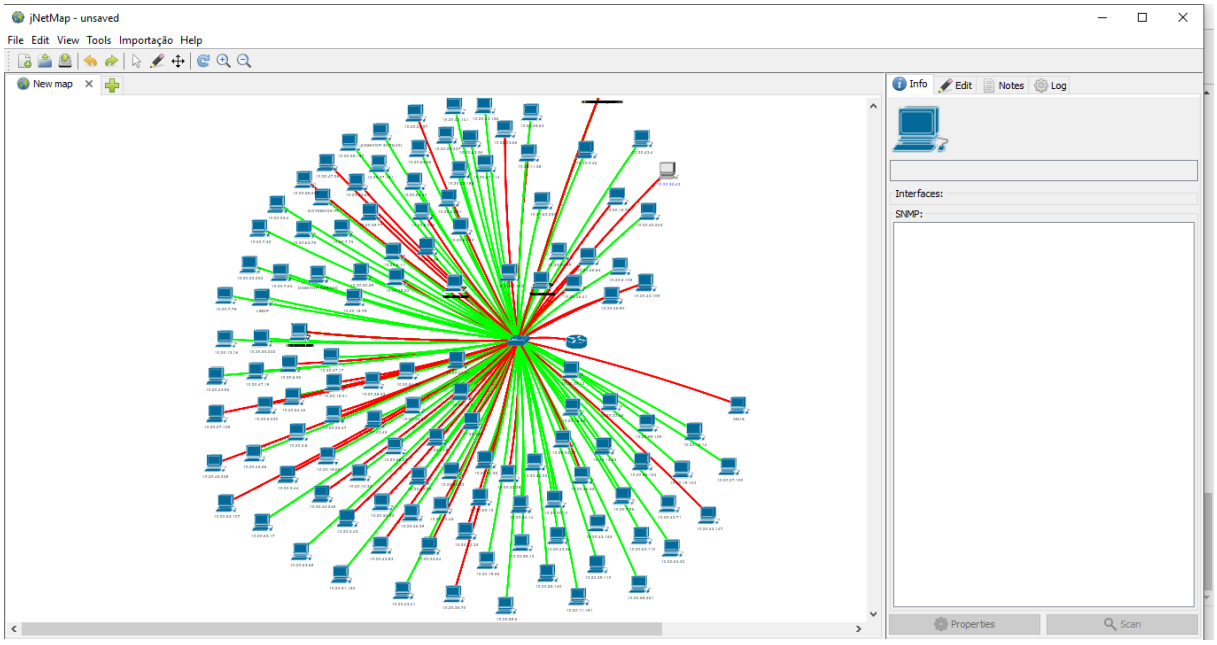
O próximo teste realizado foi com o software *jNetMap*, escaneando os dispositivos conectados na rede e criando assim um mapa da topologia de rede. Os dispositivos mapeados com a ligação na cor verde estão conectados, já os com ligação vermelha estão indisponíveis. Além disso, alguns dispositivos tiveram suas portas escaneadas. Os resultados obtidos foram:

- Rede Wi-Fi - Máquina Virtual, Windows 7: A ferramenta funcionou normalmente durante o escaneamento de dispositivos e testou com sucesso quais portas estavam sendo utilizadas pelos dispositivos.

- Rede Wi-Fi - Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, Windows 10 (notebook próprio): A ferramenta funcionou normalmente durante o escaneamento de dispositivos e testou com sucesso quais portas estavam sendo utilizadas pelos dispositivos. A Figura 44 ilustra o mapa da rede sem fio. A Figura 45 ilustra o teste de portas utilizadas por um dispositivo conectado a rede sem fio.

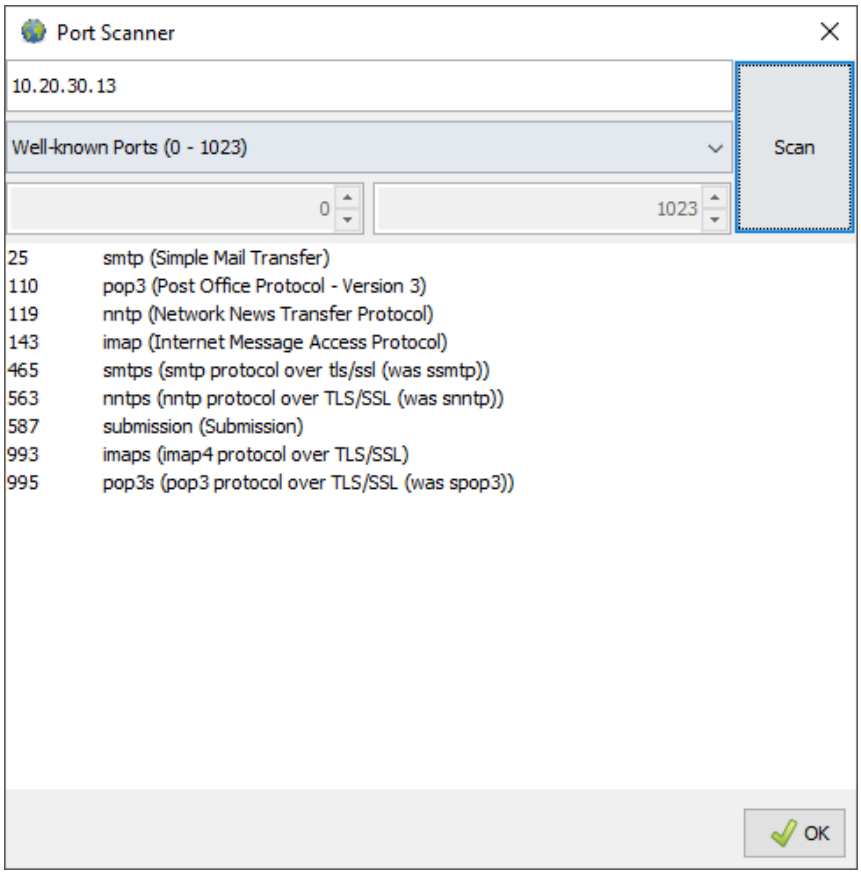
- Rede Ethernet - Laboratório de Comunicação da Área de Ciências Sociais da Universidade de Caxias do Sul, Windows 7 (desktop da UCS): A ferramenta funcionou normalmente durante o escaneamento de dispositivos e testou com sucesso quais portas estavam sendo utilizadas pelos dispositivos. A Figura 46 ilustra o mapa da rede cabeada. A Figura 47 ilustra o teste de portas utilizadas por um dispositivo conectado à rede cabeada.

Figura 44 - Mapa da rede sem fio descoberta.



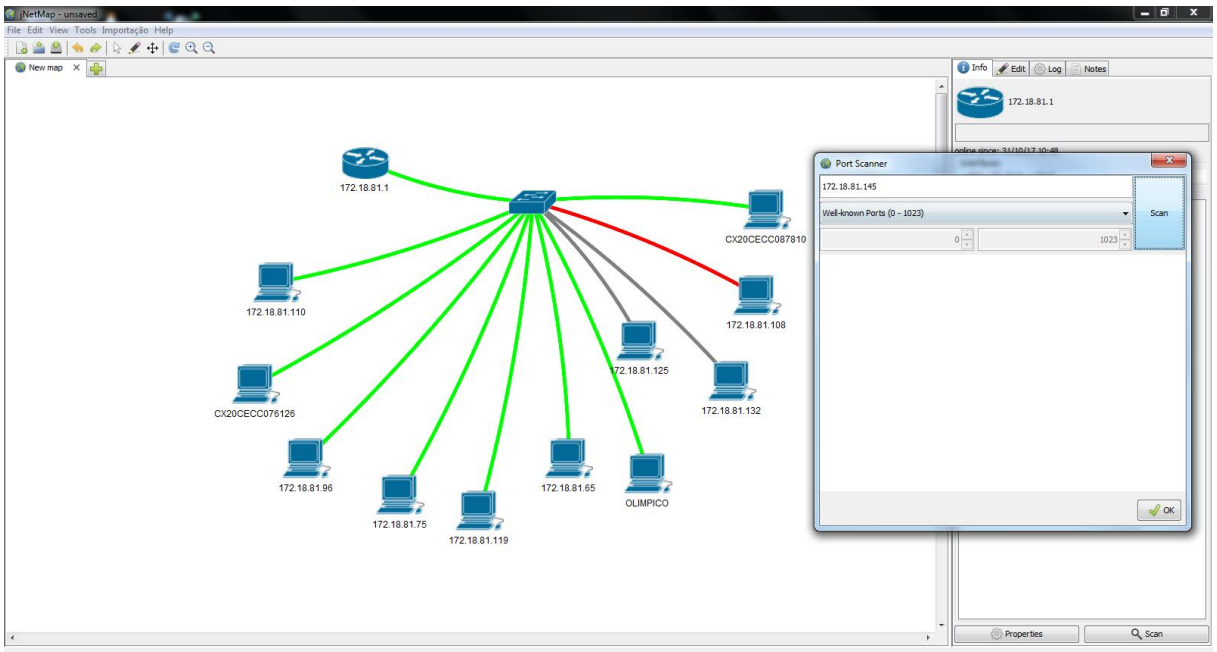
Fonte: Próprio Autor, 2017.

Figura 45 - Escaneamento de portas de um dispositivo conectado à rede sem fio.



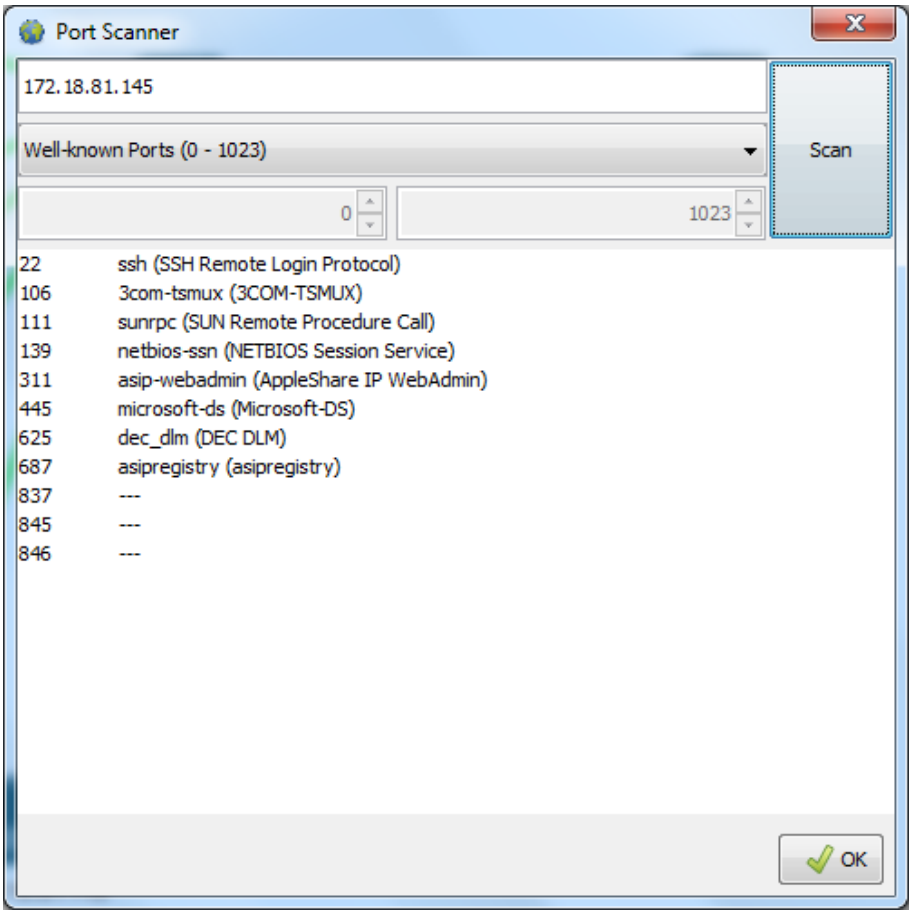
Fonte: Próprio Autor, 2017.

Figura 46 - Mapa da rede cabeada descoberta. Fonte:



Próprio Autor, 2017.

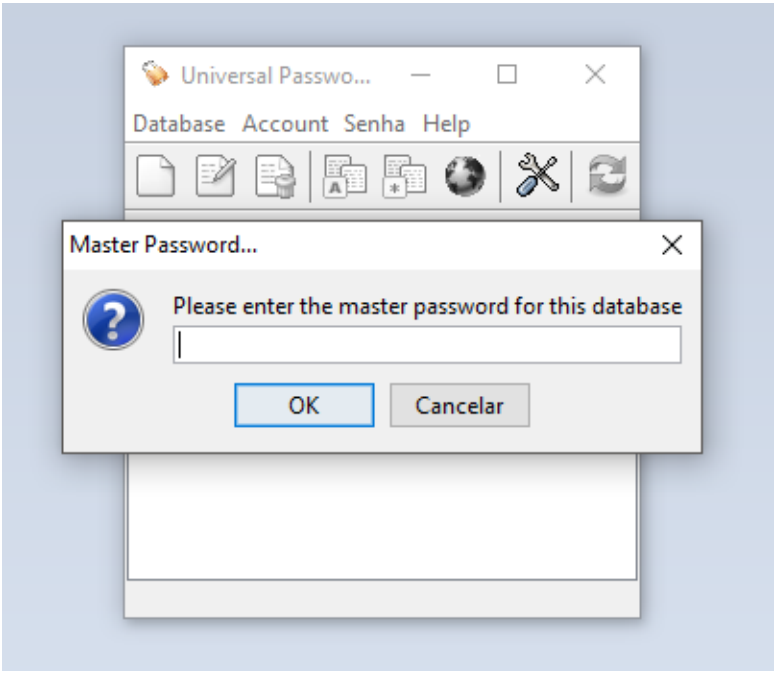
Figura 47 - Escaneamento de portas de um dispositivo conectado à rede cabeada.



Fonte Próprio Autor, 2017.

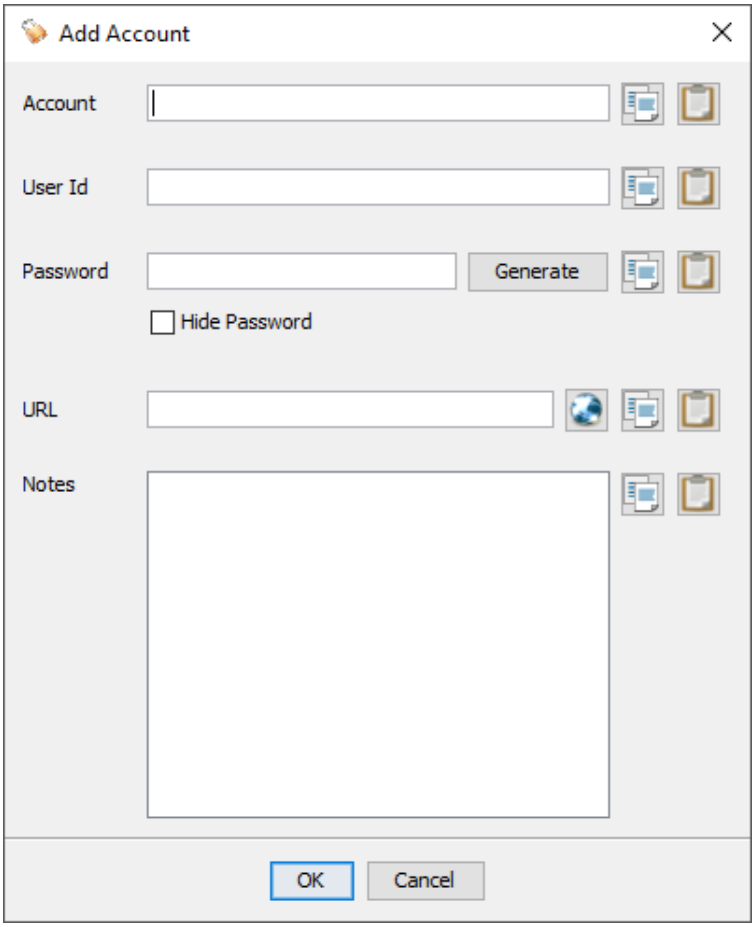
O próximo teste realizado foi criar um banco de dados criptografado na ferramenta *Universal Password Manager*, além de adicionar, remover e consultar informações de *login*. A ferramenta funcionou normalmente durante os testes nos três ambientes. A base de dados foi salva na Área de Trabalho do computador. A Figura 48 ilustra o acesso à base de dados. A Figura 49 ilustra a adição de novas credenciais de usuário ao banco de dados. A Figura 50 ilustra a visualização das credenciais salvas.

Figura 48 - Acesso ao banco de dados criptografado.



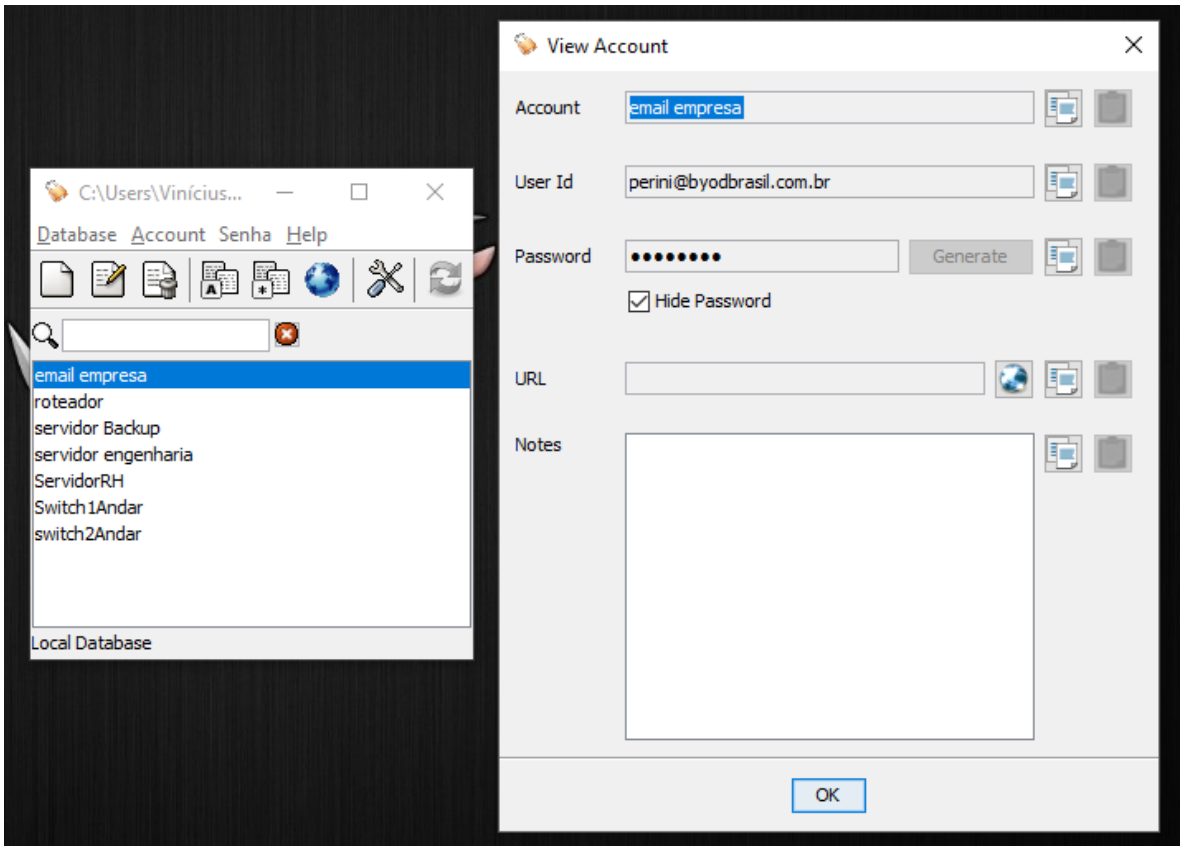
Fonte: Próprio Autor, 2017.

Figura 49 - Adicionando credenciais de usuário ao banco de dados.



Fonte: Próprio Autor, 2017.

Figura 50 - Visualizando credenciais de usuário no Universal Password Manager.



Fonte: Próprio Autor, 2017.

Após, o *software Password Strength Meter* foi executado para testar uma senha em cada nível de segurança e informar o seu respectivo número aproximado de iterações necessárias para que um algoritmo de força-bruta quebre a respectiva senha. A ferramenta funcionou normalmente durante os testes nos três ambientes. A Tabela 3 exemplifica as 16 senhas testadas e suas características.

Tabela 3 - Senhas testadas no Password Strength Meter e resultados.

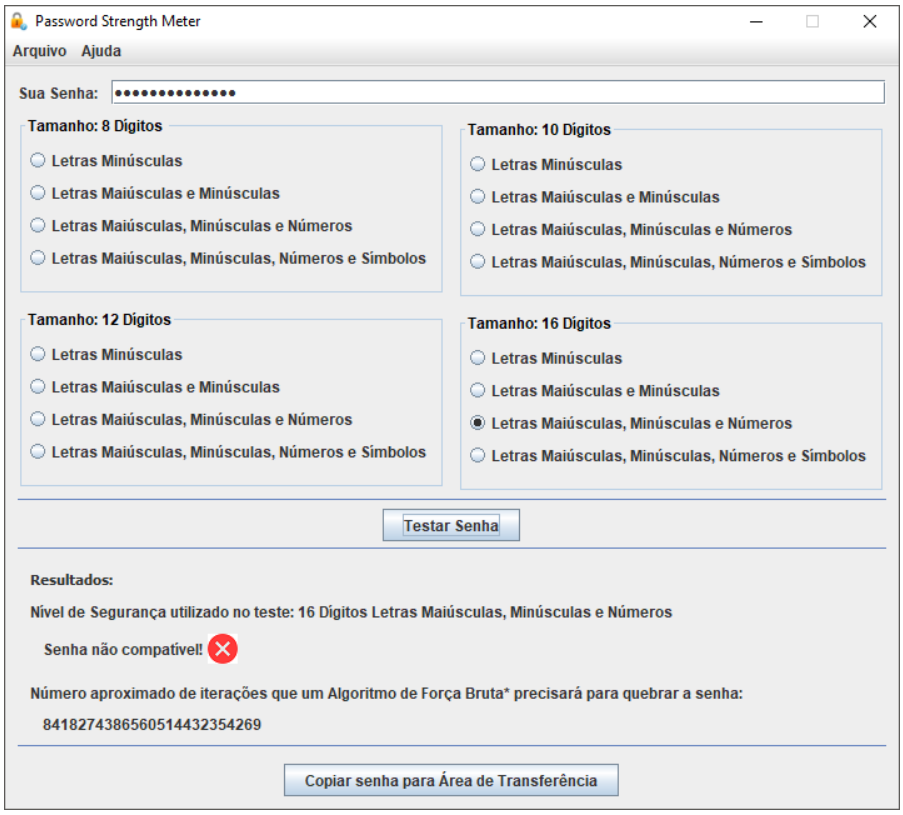
Senha	Categoria	Tamanho	Número aproximado de iterações para quebrar a senha	Tempo estimado para quebrar a senha*
abcdefgh	Somente Letras Minúsculas	8	8687205886	Instantâneo
abcDefgH	Letras Maiúsculas e Minúsculas	8	1068973602094	Instantâneo
abCdfGh42	Letras Maiúsculas, Minúsculas e Números	8	27034302717757	4 dias
Abc4jfGcv?	Letras Maiúsculas, Minúsculas, Números e Símbolos	8	17030253950511948239	10 anos
mnbdghtyzq	Somente Letras Minúsculas	10	7352451308761	59 minutos
mNbdAhtyih	Letras Maiúsculas e Minúsculas	10	38279311181248236	1 mês
mN9bdAxr61	Letras Maiúsculas, Minúsculas e Números	10	184934187100163256	8 meses
M%sksb9mab	Letras Maiúsculas, Minúsculas, Números e Símbolos	10	2511843260271183529 7	6 anos
tyrtxlmzaafd	Somente Letras Minúsculas	12	77038171375394220	4 semanas
tRWQmXpBaaFd	Letras Maiúsculas e Minúsculas	12	156836263365153567316	300 anos
Abc4jf9bdAks	Letras Maiúsculas, Minúsculas e Números	12	140671882945477366667 93	3000 anos
b9m#$Cvs679A	Letras Maiúsculas, Minúsculas, Números e Símbolos	12	150968938214621507975 42	34000 anos
mnbdaafeficnkase	Somente Letras Minúsculas	16	22712858962317404722639	35000 anos
MjXkioZfeWghxTYa	Letras Maiúsculas e Minúsculas	16	21550468950770499718122 6657	2 bilhões de anos
iotrAa13fghxxTwe	Letras Maiúsculas, Minúsculas e Números	16	71102739361554856249042 7087	38 bilhões de anos
B1daXZeC2Fhx@#$e	Letras Maiúsculas, Minúsculas, Números e Símbolos	16	13235705097905705563866635469955	1 trilhão de anos

* O tempo estimado foi calculado no site: <https://howsecureismypassword.net/>. O algoritmo utilizado nesta determinação e sua documentação estão disponíveis em: <https://github.com/howsecureismypassword/hsimp>. Este algoritmo contém um dicionário com as senhas mais utilizadas no mundo.

Fonte: Próprio Autor, 2017.

A Figura 51 ilustra a interface de usuário do *Password Strength Meter* testando uma senha.

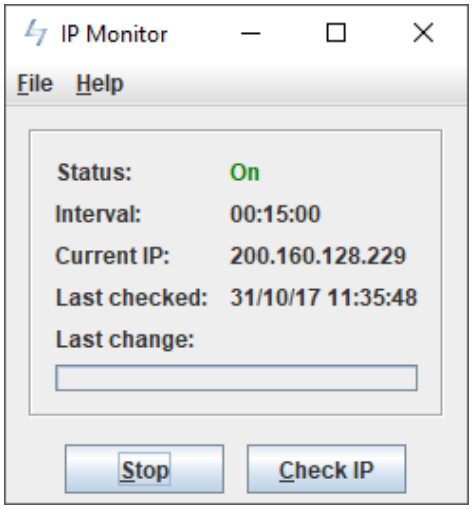
Figura 51 - Testando uma senha no Password Strength Meter.



Fonte: Próprio Autor, 2017.

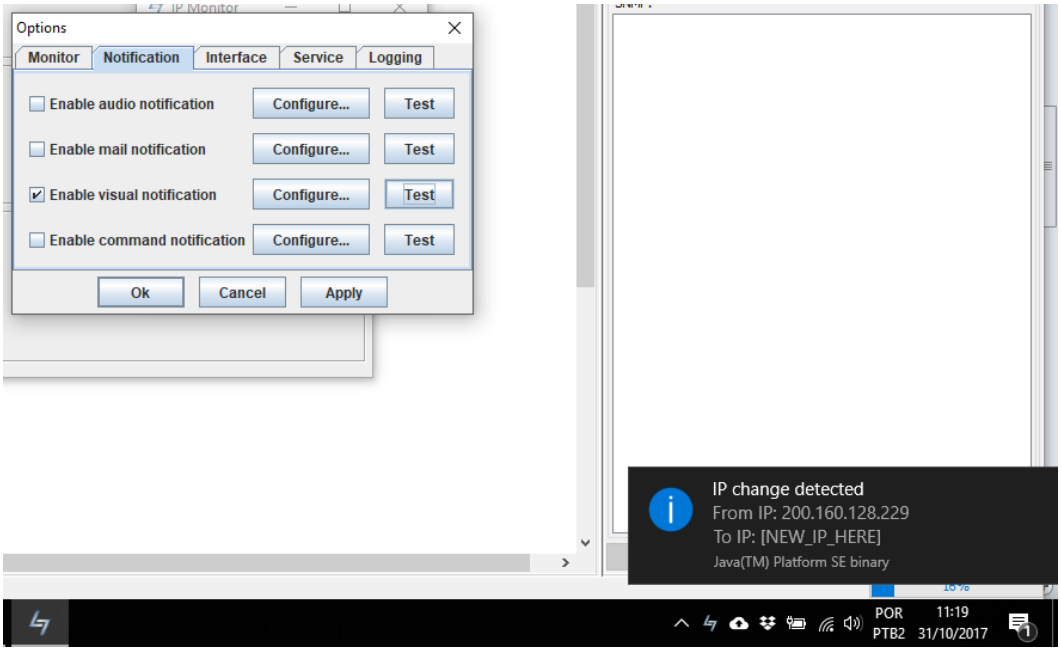
O *software IP Monitor* foi utilizado para monitorar o endereço de IP público durante quinze minutos e testar uma opção de notificação. A ferramenta funcionou normalmente durante os testes nos três ambientes. A Figura 52 ilustra a interface principal do *IP Monitor*, durante o período de escaneamento. A Figura 53 ilustra o teste de notificação visual do *IP Monitor*, onde é possível ver o sistema nativo de notificações do Windows 10 funcionando.

Figura 52 - Monitoramento do endereço de IP público.



Fonte: Próprio Autor, 2017.

Figura 53 - Notificação visual da troca de endereço IP público.

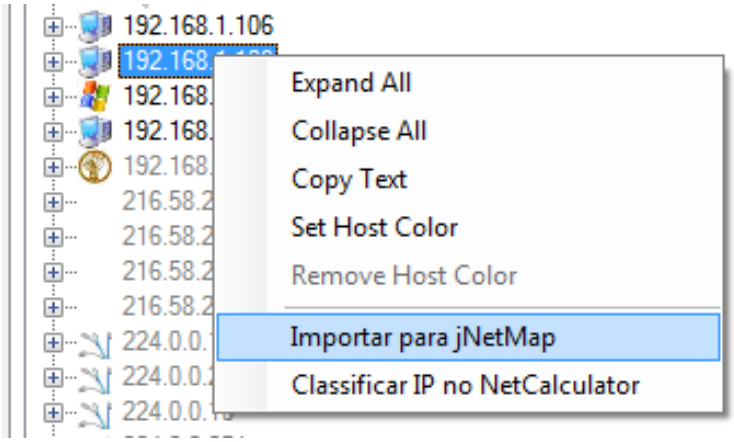


Fonte: Próprio Autor, 2017.

6.2 TESTES DE FUNCIONALIDADES DA INTEGRAÇÃO

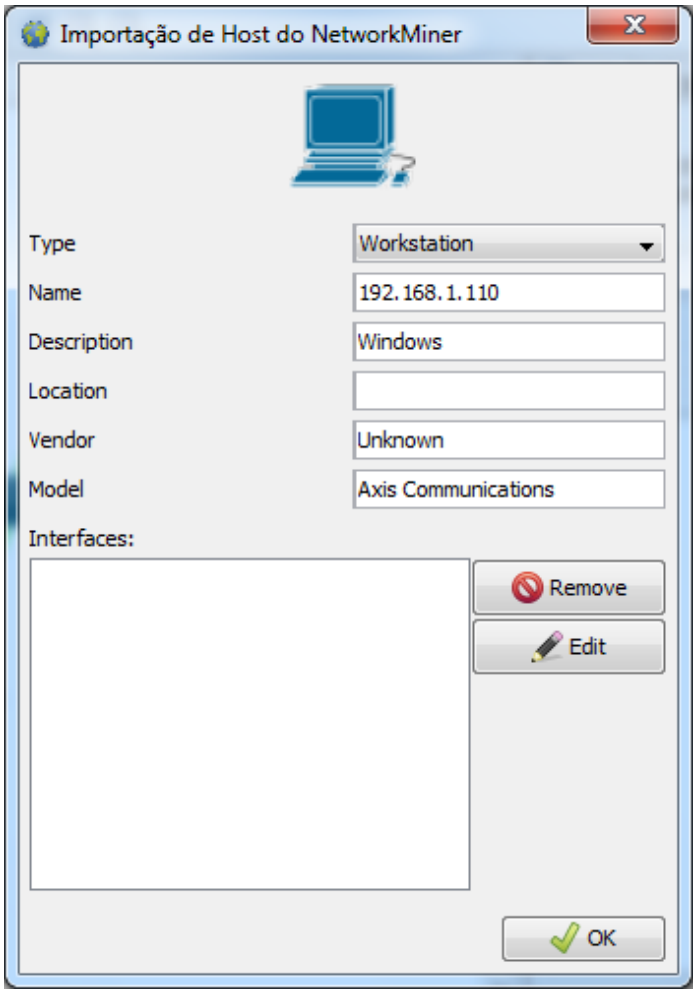
O primeiro teste realizado foi importar diversos hosts do *NetworkMiner* para o *jNetMap*. Nos ambientes que utilizam a rede sem fio, as ferramentas funcionaram normalmente durante as importações. A Figura 54 ilustra a opção "Importar para *jNetMap*" disponível no *NetworkMiner* e a Figura 55 ilustra o recebimento do host pelo *jNetMap*. Na Figura 55 é possível notar alguns campos preenchidos automaticamente com os dados recebidos do *NetworkMiner*. No ambiente da rede cabeada, conforme mencionado na seção 6.1, o *NetworkMiner* apresentou problemas em decorrência do antivírus instalado no ambiente que bloqueou a sua execução.

Figura 54 - Importando um host do NetworkMiner para o jNetMap.



Fonte: Próprio Autor, 2017.

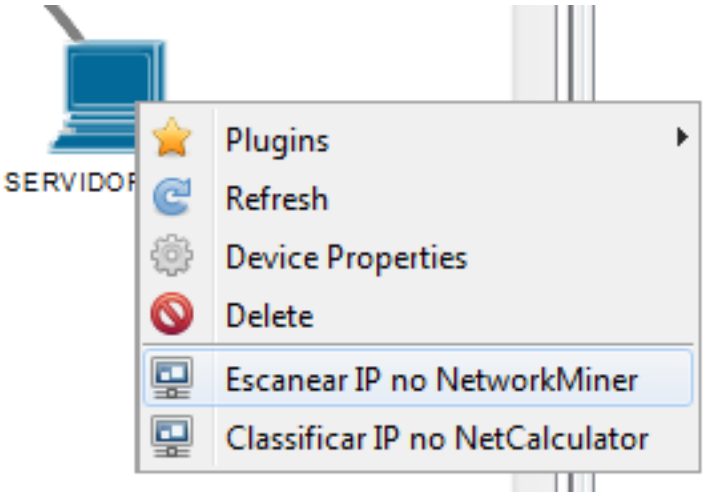
Figura 55 - Importação e configuração de host no jNetMap.



Fonte: Próprio Autor, 2017.

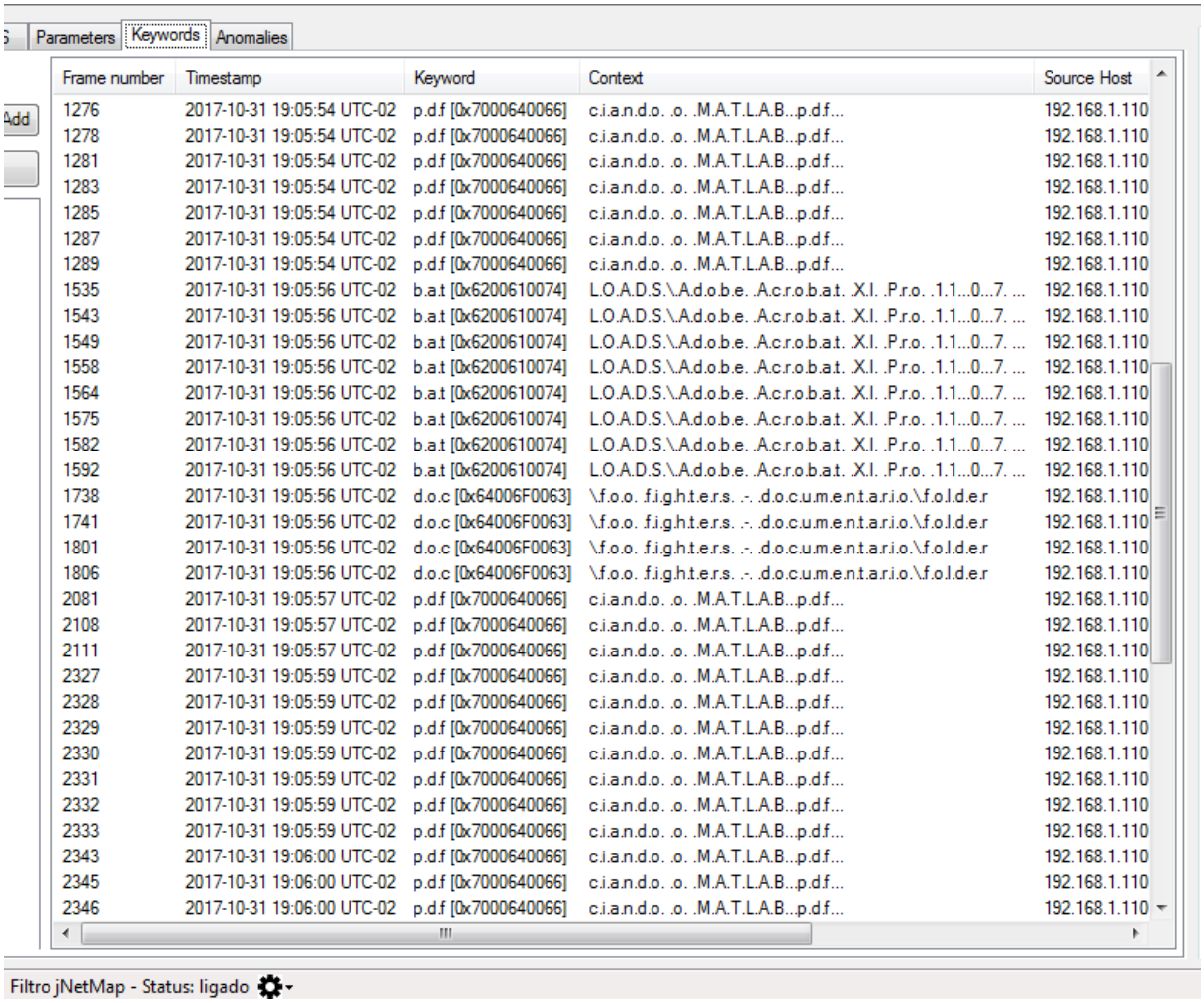
Após, os resultados obtidos pelo escaneamento do *NetwokMiner* foram filtrados através de um endereço IP previamente enviado pelo *jNetMap*. Nos ambientes de rede sem fio, ambas ferramentas funcionaram normalmente durante a troca de dados. A Figura 56 ilustra a opção "Escanear IP no *NetworkMiner*". A Figura 57 ilustra os resultados filtrados pelo endereço "192.168.1.103" pertencente ao dispositivo "SERVIDOR-DLINK" que foi enviado pelo *jNetMap*. Na Figura 57 é possível ver arquivos com formato "pdf", "doc" e "bat". A visualização destes arquivos foi possível porque foram utilizadas as mesmas *keywords* de escaneamento da seção 6.1. Estes arquivos estavam armazenados no endereço "192.168.1.103" e foram acessados pela máquina virtual com endereço "192.168.1.110". Na rede Ethernet, a ferramenta NetworkMiner apresentou o mesmo problema devido ao antivírus.

Figura 56 – Enviando um endereço IP do jNetMap para o NetworkMiner.



Fonte: Próprio Autor, 2017.

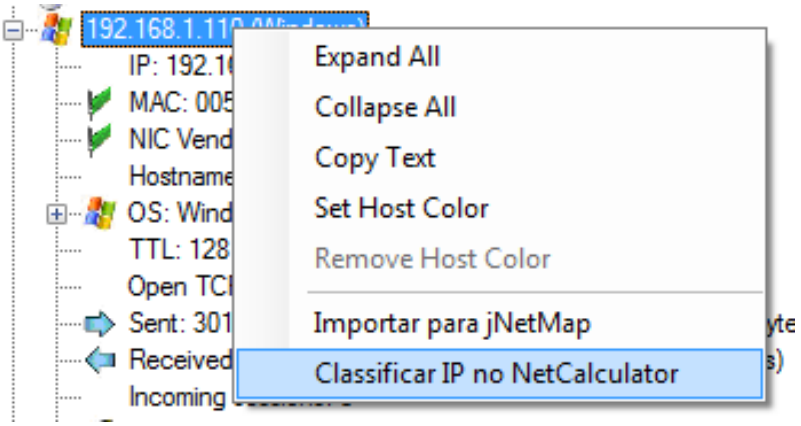
Figura 57 - Resultados do escaneamento do NetworkMiner (filtro ligado).



Fonte: Próprio Autor, 2017.

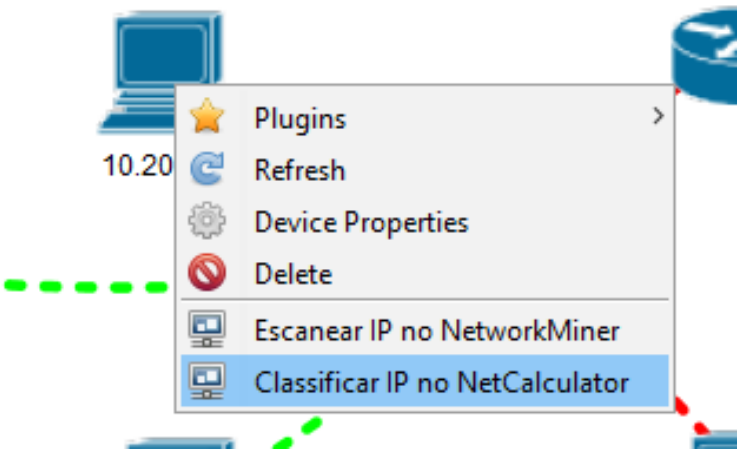
Finalmente, os endereços foram classificados no NetCalculator. Os endereços foram enviados a partir do *NetworkMiner* e do *jNetMap*. Nos ambientes que utilizam a rede sem fio, ambas ferramentas funcionaram normalmente durante a classificação de endereços. A Figura 58 ilustra a opção "Classificar IP no *NetCalculator*". A Figura 60 ilustra a classificação do IP "192.168.1.110" que foi recebido. A Figura 59 ilustra a opção "Classificar IP no *NetCalculator*" no *jNetMap*. A Figura 61 ilustra a classificação do IP "10.20.0.11"que foi recebido. Conforme mencionado na seção 6.1, o *NetworkMiner* apresentou problemas relacionados ao antivírus no ambiente que utiliza a rede cabeada.

Figura 58 - Enviando um IP para o NetCalculator a partir do NetworkMiner.



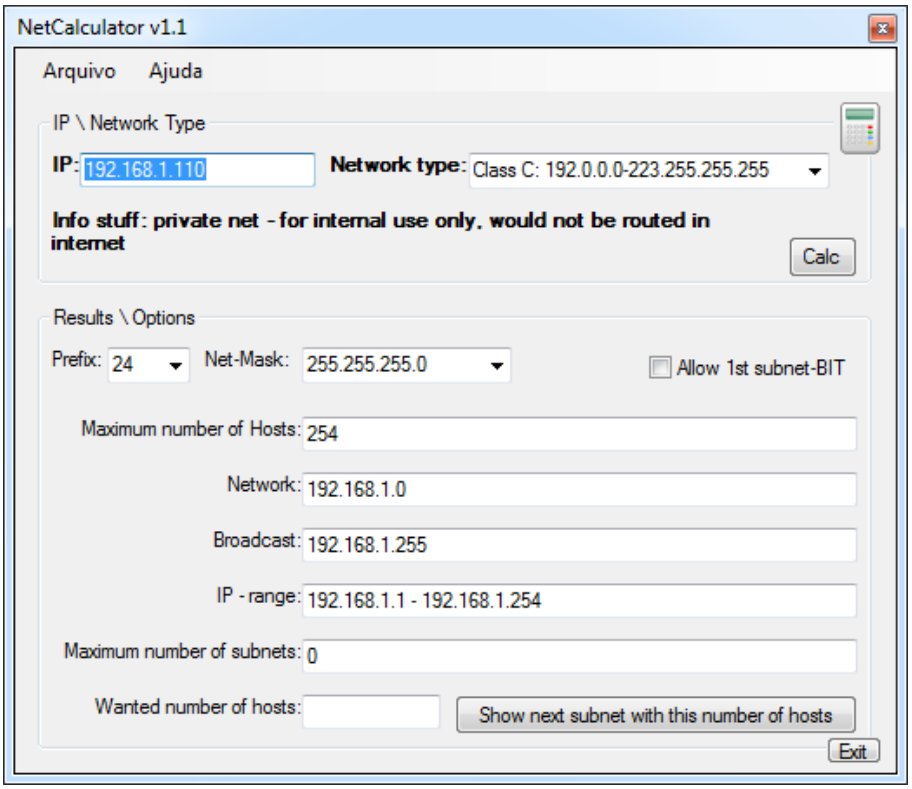
Fonte: Próprio Autor, 2017.

Figura 59 - Enviando um endereço IP a partir do jNetMap.



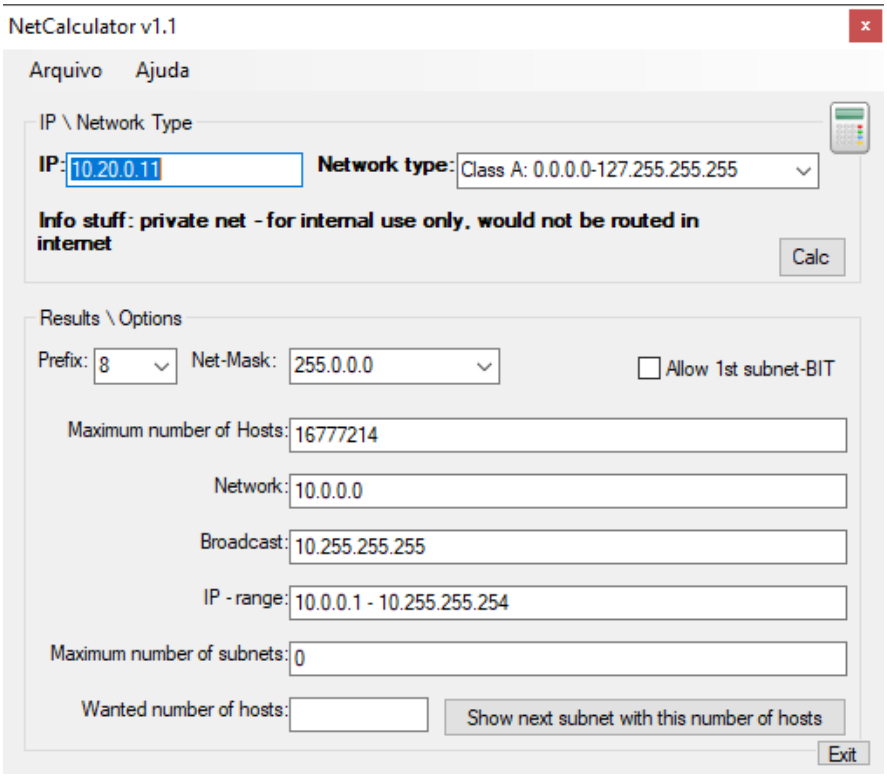
Fonte: Próprio Autor, 2017.

Figura 60 - Classificando IP recebido do NetworkMiner.



Fonte: Próprio Autor, 2017.

Figura 61 - Classificando IP recebido do jNetMap.



Fonte: Próprio Autor, 2017

O próximo teste realizado foi analisar algumas senhas salvas no *Universal Password Manager* através da opção "*Password Strength Meter*" no menu "Senha" que foi integrado. A ferramenta funcionou normalmente durante os testes nos três ambientes. Os resultados obtidos foram semelhantes aos valores da Tabela 3, citada anteriormente.

6.3 TESTES DAS CATEGORIAS DE VULNERABILIDADES

Nesta etapa, as ferramentas foram testadas de acordo com as quatro grandes categorias de vulnerabilidades BYOD, citadas na seção 2.2. Possíveis situações reais foram criadas para que fosse possível analisar o comportamento das ferramentas.

O primeiro teste realizado foi verificar se as portas que estão sendo utilizadas por um dispositivo são portas comumente utilizadas por *malwares*. Além disso, foram testados endereços IPs que possam indicar um tráfego suspeito. Em caso positivo, isso significaria uma possibilidade de ameaça à rede BYOD. Para realizar este comparativo, duas listas foram utilizadas como fonte: a lista de portas[18] do *Trend Micro*[19] e a lista de endereços do *SpyBot*[20]. Este teste está relacionado diretamente com a primeira categoria de vulnerabilidade: *malwares*.

O teste funcionou corretamente nos dois ambientes que utilizam a rede sem fio. O ambiente com rede *Ethernet*, como já citado, apresentou os mesmos problemas na execução do *NetworkMiner*. Em nenhum ambiente, o *SpyBot* estava instalado. A partir da lista de endereços do *SpyBot*, o site "www.007guard.com" foi escolhido para ser testado. A Figura 62 ilustra, no primeiro ambiente (máquina virtual), a captura de pacotes de rede que se originaram do site escolhido. A captura foi feita utilizando o *NetworkMiner*. Neste momento, o profissional de TI pode tomar as medidas de segurança, de acordo com as políticas da empresa, como por exemplo, bloquear o acesso ao site.

---

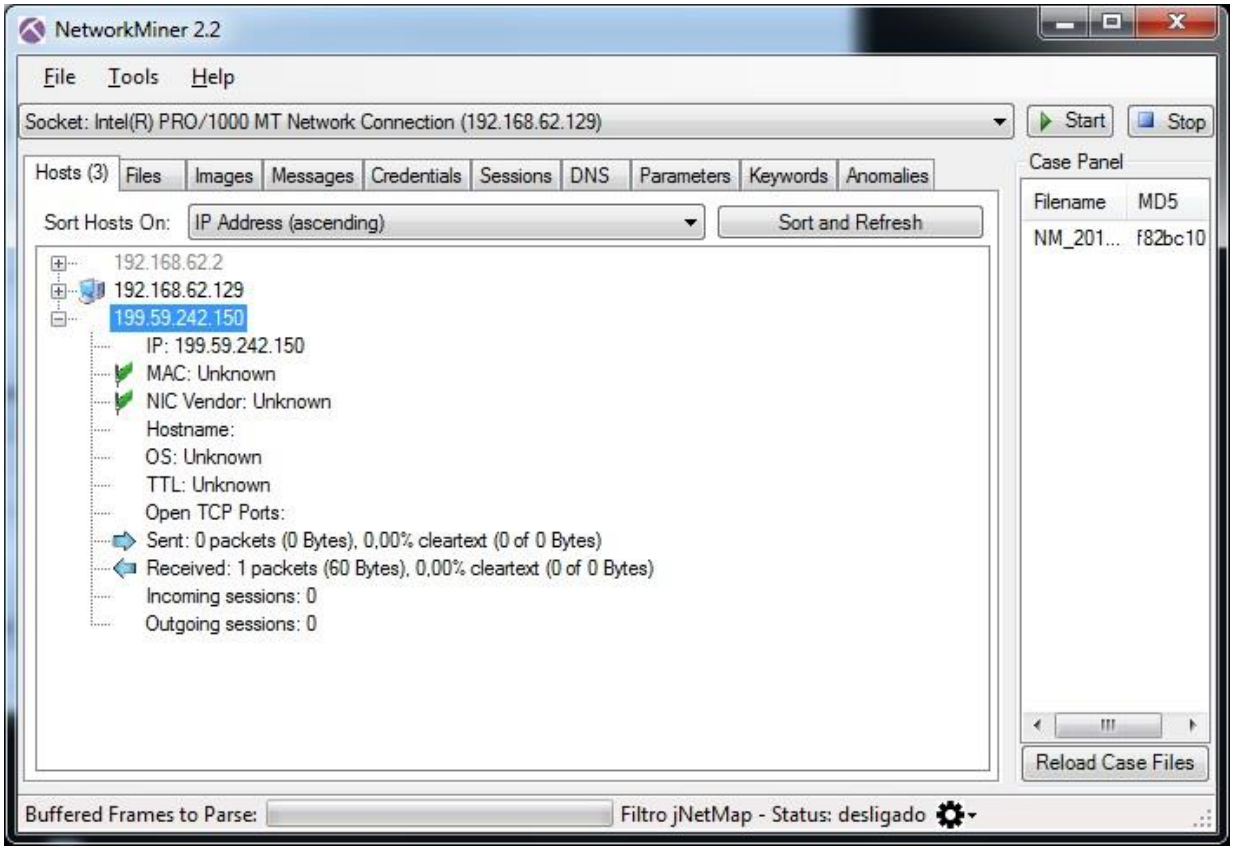
[18] A lista do Trend Micro é um compilado disponibilizado pela própria Trend Micro que reúne as portas mais comumente utilizadas por *malwares*. A tabela está disponível em: <http://docs.trendmicro.com/all/ent/officescan/v10.5/en-us/osce_10.5_olhsrv/osceag/osceag-appa/trojan_port.htm>.

[19] Uma lista de URLs e domínios de IP maliciosos é disponibilizada pela Trend Micro e está disponível em: <http://www.trendmicro.com.br/br/inteligencia-de-seguranca/atividade-atual-de-ameacas/dez-mais-maliciosos/index.html>

[20] O SpyBoy é um *software* desenvolvido para eliminar *malwares*, *spywares* e *adwares*. Durante seu funcionamento, o SpyBot cria uma lista de endereços (sites maliciosos) no arquivo "hosts" do Windows. Este arquivo fica no diretório "C:/System32/drivers/etc/" e é utilizado pelo sistema operacional como um complemento ao DNS. Editando este arquivo, o SpyBot é capaz de evitar com que certos sites sejam acessados pelo computador, retornando sempre o endereço "127.0.0.1".

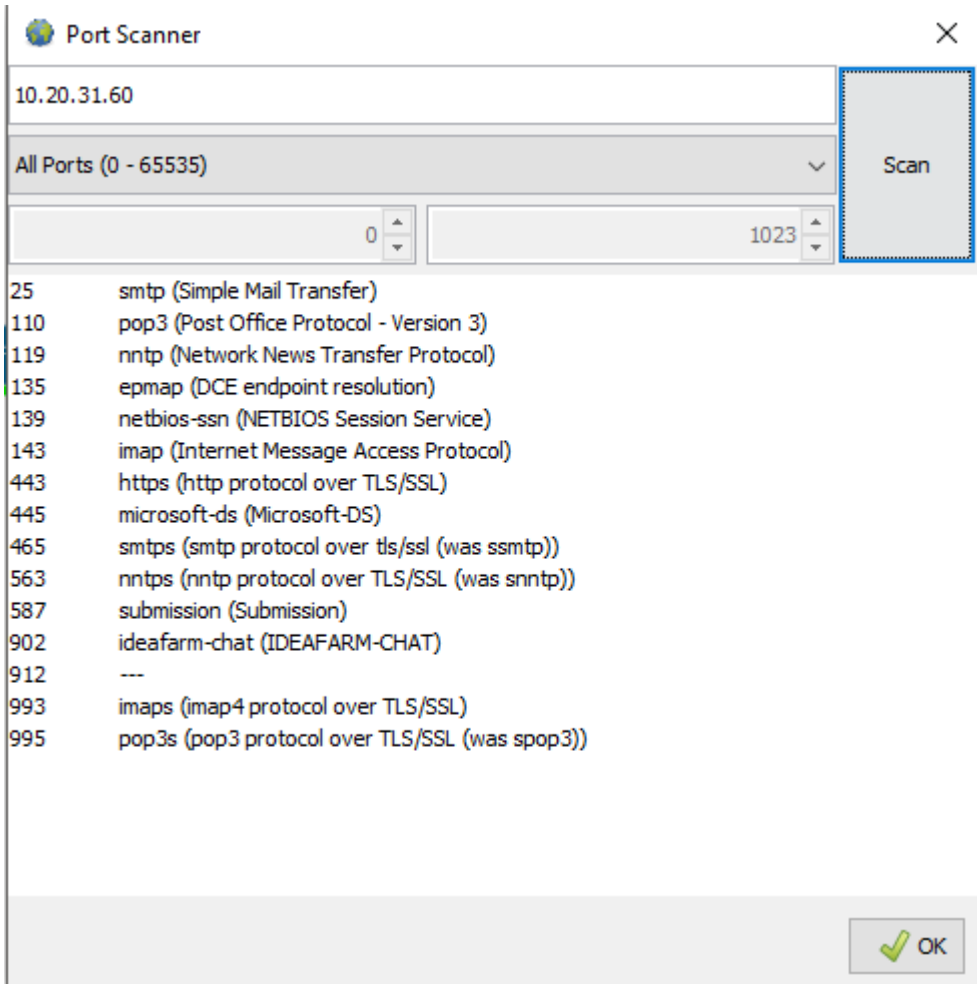
Já o teste de portas em utilização foi feito no segundo ambiente (rede sem fio) utilizando o *jNetMap*. A opção "*All Ports*" foi selecionada para que todas as portas existentes fossem testadas no dispositivo em questão. O resultado dos testes está ilustrado na Figura 63. De acordo com a lista de portas do *Trend Micro*, nenhuma possibilidade de ameaça foi descoberta.

Figura 62 - Capturando pacotes de endereço IP suspeito.



Fonte: Próprio Autor, 2017.

Figura 63 - Escaneamento completo de portas de um dispositivo.



Fonte: Próprio Autor, 2017.

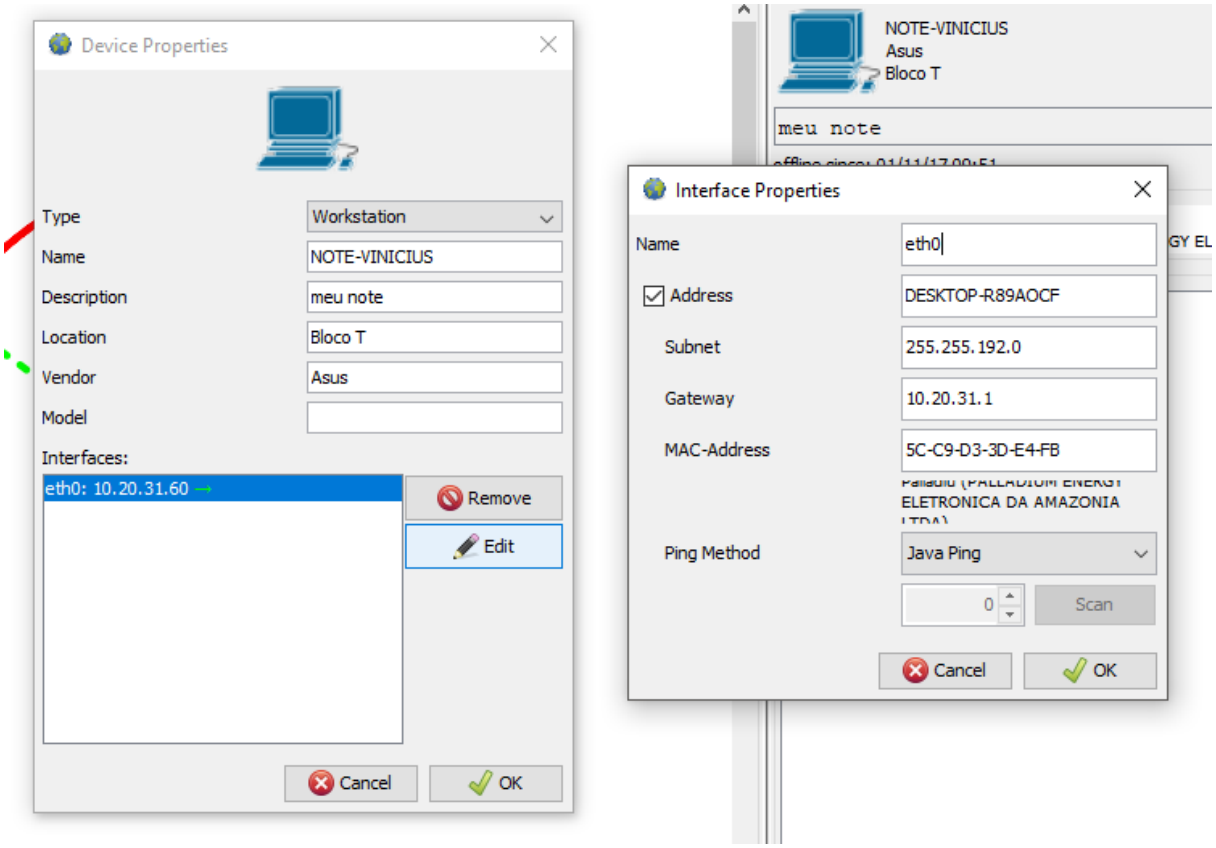
Outra vulnerabilidade testada refere-se à permissão de usuário. Para realizar este teste, foi necessário supor um ambiente empresarial no qual o administrador de rede configurou níveis de acesso através de endereços MAC. Os dispositivos que pertencem aos colaboradores da empresa tiveram seus endereços MAC previamente cadastrados. Desta forma, foi possível verificar se existe algum dispositivo conectado à rede que não possui o endereço MAC cadastrado.

O profissional de TI poderá consultar o endereço MAC de um dispositivo através do *jNetMap* ou do *NetworkMiner*. O teste funcionou corretamente nos dois ambientes com rede sem fio. O ambiente que utiliza a rede cabeada, como já citado, apresentou problemas na execução do *NetworkMiner*. Para verificar os dispositivos conectados à rede, utilizando o *jNetMap*, é necessário utilizar a ferramenta "*Network Scan*", disponível no menu "*Tools*". Após a execução, os dispositivos que estão conectados à rede serão adicionados ao mapa.

Caso o profissional de TI note algum dispositivo diferente, é possível consultar seus dados através da opção "*Device Properties*" clicando com o botão direito sobre o dispositivo. A Figura 64 ilustra a consulta de um dispositivo já cadastrado. A Figura 65 ilustra o campo "*MAC-Address*" em branco, isso significa que um dispositivo novo está conectado à rede. Neste momento o profissional de TI pode realizar os procedimentos padrões de segurança, como por exemplo, desconectar este dispositivo da rede, ou realizar uma captura de pacotes com o *NetworkMiner*.

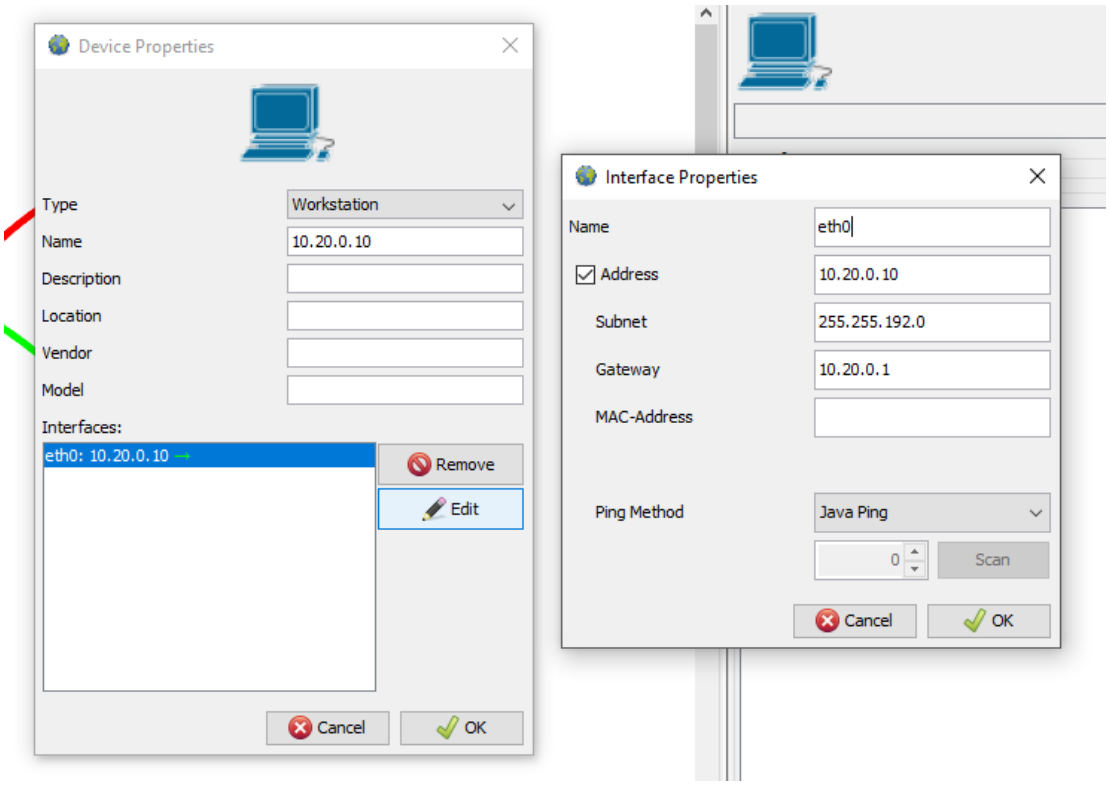
A Figura 66 ilustra a captura de pacotes filtrando somente resultados relacionados ao IP que não possui um endereço MAC previamente cadastrado. Caso seja um dispositivo de um funcionário da empresa, o mesmo pode entrar em contato com o setor de tecnologia e solicitar o registro de seu dispositivo móvel.

Figura 64 - Consulta detalhada de propriedades do dispositivo no jNetMap.



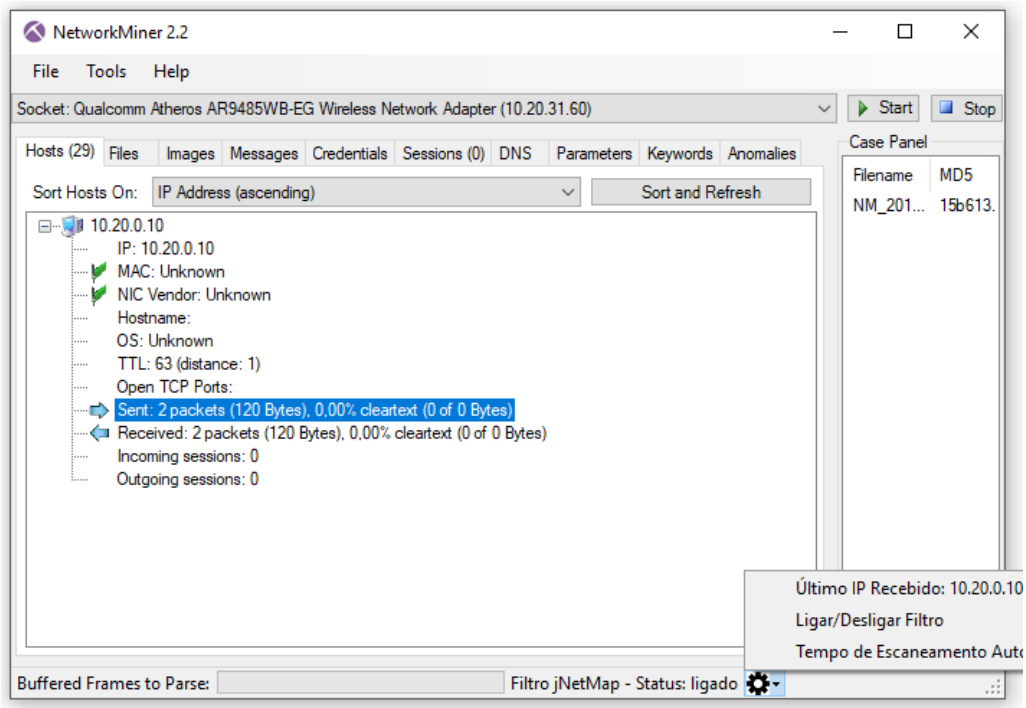
Fonte: Próprio Autor, 2017.

Figura 65 - Consulta detalhada de propriedades do dispositivo no jNetMap.



Fonte: Próprio Autor, 2017.

Figura 66 - Resultado da captura de pacotes filtrando um dispositivo específico.

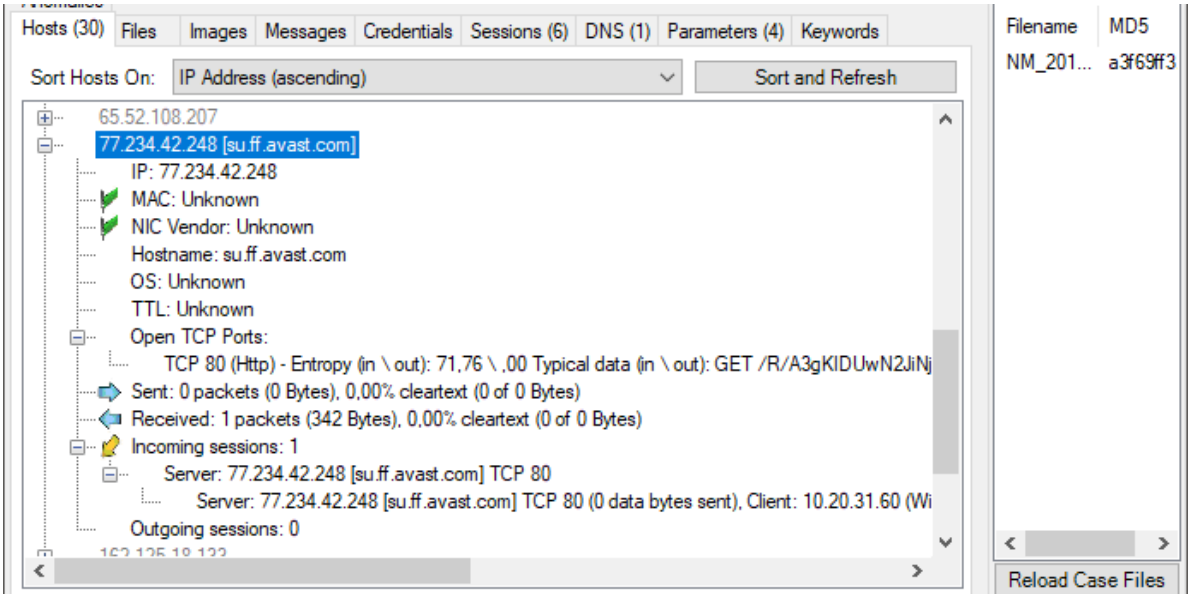


Fonte: Próprio Autor, 2017.

O último teste realizado foi: analisar tráfego suspeito e possível quebra de senha da rede sem fio. Caso senhas estejam sendo trafegadas sem um nível de criptografia seguro, é possível que terceiros estejam interceptando estes dados. A análise deve ser feita comparando o número de pacotes recebidos e enviados por um host com os demais hosts. Além disso, é possível comparar as "*Incoming Sessions*" e "*Outcoming Sessions*". Este teste está relacionado diretamente com a terceira categoria de vulnerabilidade: encriptação.

O teste funcionou corretamente nos dois primeiros ambientes. No terceiro ambiente, o antivírus *Trend Micro* bloqueou a execução do *NetworkMiner*. Nenhum host apresentou dados alarmantes durante o período de captura de pacotes. Em caso positivo, uma quantidade expressiva de dados poderia sinalizar um roubo de informações sensíveis. Neste caso, o profissional de TI pode aplicar alguma medida de segurança de acordo com as políticas da corporação. A Figura 67 ilustra detalhadamente quais portas estão sendo utilizadas pelo *host*, quantos pacotes foram recebidos e enviados, e quantas "*Incoming Sessions*" e "*Outcoming Sessions*" foram capturadas.

Figura 67 - Detalhes de tráfego de um host no NetworkMiner.



Fonte: Próprio Autor, 2017.

A quarta e última categoria de vulnerabilidade é a desatenção do usuário. Esta categoria não pode ser testada no *software BYOD Manager Kit* porque é um erro humano.

7. CONCLUSÃO

A inclusão digital está facilitando o acesso da população, em todo o mundo, à internet e outras tecnologias. O estudo realizado mostrou como as empresas estão se modernizando e adotando novas formas de trabalho. Uma modalidade de trabalho que está em crescimento e vem apresentando bons resultados é o modelo BYOD. Empresas que adotaram este modelo constataram uma economia nos investimentos em tecnologia e infraestrutura e os funcionários ainda obtiveram melhores resultados.

O modelo BYOD exige algumas mudanças na forma como a empresa administra e garante a segurança da informação. Os *softwares* utilizados pelo profissional de TI devem estar de acordo com as normas de segurança da empresa, além de garantir a compatibilidade com diversos tipos de dispositivos e diversos sistemas operacionais. O protótipo de *software* desenvolvido reuniu seis ferramentas diferentes, que possuem diferentes objetivos finais, com o intuito de criar uma única suíte de ferramentas chamada BYOD Manager Kit.

Softwares de administração e segurança de redes são extremamente necessários por diversos fatores. Alguns exemplos foram citados ao longo do Capítulo 2 (BYOD – *Bring Your Own Device*), mas é necessário estar em constante processo de atualização. A cada semana novos problemas de segurança são descobertos e alguns deles podem ter proporções globais, como aconteceu recentemente com o *ransomware WannaCry*[21], que causou prejuízos incontáveis e fez boa parte dos computadores do mundo ficarem inutilizáveis.

Muitas vezes o funcionário da empresa não tem conhecimento técnico para lidar com estes problemas, então o profissional de TI deve aumentar a atenção a qualquer suspeita na rede BYOD. Campanhas informativas e de treinamento são boas opções para diminuir as falhas humanas na segurança de redes.

O BYOD Manager Kit é um protótipo que pode auxiliar o profissional de TI na administração e segurança de redes BYOD. Conforme foi explícito no Capítulo 6 (Testes e Resultados), a solução proposta atendeu às diversas necessidades e as novas funcionalidades foram de grande valia para o projeto. As ferramentas de análise e descobrimento de redes são indispensáveis no dia a dia do administrador de rede. Elas possibilitam monitorar o que está sendo transmitido pela rede, incluindo formatos específicos de arquivos, além de possibilitar a

[21] O *WannaCry* foi um *ransomware* que se espalhou por mais de 150 países de maneira extremamente rápida. Ele criptografava os arquivos do dispositivo infectado e exigia um pagamento em criptomedas para supostamente liberar o acesso aos arquivos. Um artigo foi publicado pela *International Journal of Advanced Research in Computer Science* e pode ser acessado em: <http://www.ijarcs.info/index.php/Ijarcs/article/view/4021>.

visualização gráfica de dispositivos conectados à rede. Em um ambiente corporativo, estas ferramentas podem ser utilizadas para evitar que informações confidenciais não sejam roubadas por terceiros.

Algumas melhorias podem ser realizadas futuramente neste projeto. Em decorrência do curto espaço de tempo não foi possível adicionar outras funcionalidades à integração. Por exemplo, quando uma senha é testada no *Password Strength Meter*, a biblioteca informa a quantidade aproximada de iterações que um algoritmo de força-bruta precisa para quebrar esta senha. Só esta informação pode não ser o suficiente para que o administrador de rede saiba se a senha é forte o suficiente. Seria interessante adicionar o tempo (anos, meses, dias, horas, minutos e segundos) necessário para que um computador atual quebre esta senha.

O protótipo foi desenvolvido em linguagem Java, pois isso facilita o funcionamento em diversos ambientes. Uma versão do BYOD Manager Kit poderia ser desenvolvida para o Linux com algumas modificações. O *NetworkMiner*, apesar de ser desenvolvido em C#, possui uma versão Linux com documentação disponível no site oficial. Esta mesma documentação facilitaria a execução do *NetCalculator*, também desenvolvido em C#, no ambiente Linux.

Novos estudos podem ser desenvolvidos a partir deste projeto. Como o BYOD ainda é um tema relativamente novo, a quantidade de artigos e estudos sobre o assunto é relativamente pequena. *Softwares* que possibilitam o gerenciamento remoto de dispositivos, como a ferramenta *Mobi Control*, tendem a ser mais requisitados por corporações.

Um novo estudo, com base neste projeto, já está em desenvolvimento pelo aluno Marcelo Vinícius Zanol, do curso Sistemas de Informação da Universidade de Caxias do Sul. O projeto do Marcelo vai adicionar mais ferramentas *open-source* ao BYOD Manager Kit, possibilitando uma melhor abrangência. Consequentemente, o *software* vai atender novas necessidades e poderá auxiliar ainda mais o profissional de TI responsável pela administração e segurança de redes BYOD.

REFERÊNCIAS BIBLIOGRÁFICAS

Aircrack-ng. Disponível em: < https://www.aircrack-ng.org/ >. Acesso em: 22 Maio 2017.

Cegali, Fábio. Segurança digital e os desafios da nova era da Internet das Coisas, 2017. Disponível em: <http://computerworld.com.br/seguranca-digital-e-os-desafios-da-nova-era-da-internet-das-coisas> Acesso em: 09 Março 2017.

Celestino, André Luis. O conceito e as dúvidas sobre o MVC, 2014. Disponível em: <https://www.profissionaisti.com.br/2014/10/o-conceito-e-as-duvidas-sobre-o-mvc/>. Acesso em: 13 Junho 2017.

CIO. Virtualização de aplicações: você ainda vai adotar, 2011. Disponível em: <http://cio.com.br/tecnologia/2011/09/12/virtualizacao-de-aplicacoes-voce-ainda-vai-adotar/>. Acesso em: 27 Março 2017.

CIO. Proteção de dados em dispositivos móveis preocupa mais do que ciberataques, 2016. Disponível em: <http://cio.com.br/noticias/2016/11/25/protecao-de-dados-em-dispositivos-moveis-procupa-mais-do-que-ciberataques/> Acesso em: 10 Março 2017.

Computer World. Complexidade corporativa coloca em risco a segurança da informação, 2017. Disponível em: <http://computerworld.com.br/complexidade-corporativa-coloca-em-risco-seguranca-da-informacao-diz-estudo> Acesso em: 09 Março 2017.

Computer World. 3,8 milhões de smartphones são vendidos no mundo diariamente, 2016. Disponível em: < http://computerworld.com.br/38-milhoes-de-smartphones-sao-vendidos-no-mundo-diariamente>. Acesso em: 28 Março 2017.

Constantin, Lucian. Deficiências contra ameaças em dispositivos de IoT podem afetar redes locais, 2017. Disponível em: <http://computerworld.com.br/deficiencias-contra-ameacas-em-dispositivos-de-iot-podem-afetar-redes-locais> Acesso em: 10 Março 2017.

De La Merced, Michael J. Cisco to Buy Sourcefire, a Cybersecurity Company, for $2.7 Billion, 2013. Disponível em: <https://dealbook.nytimes.com/2013/07/23/cisco-to-buy-sourcefire-a-cybersecurity-company-for-2-7-billion/?_r=0>. Acesso em: 21 Maio 2017.

Delaney, Darragh. Remote access technologies in a BYOD era, 2012. Disponível em: <http://www.computerworld.com/article/2472058/infrastructure-management/remote-access-technologies-in-a-byod-era.html>. Acesso em: 04 Abril 2017.

Earls, Alan. Closing the gate: Data leak prevention, 2015. Disponível em: <https://www.scmagazine.com/closing-the-gate-data-leak-prevention/article/536721/>. Acesso em: 27 Março 2017.

Eclipse. Eclipse Wiki, 2017. Disponível em: <https://wiki.eclipse.org/Eclipse/Installation>. Acesso em: 30 Maio 2017.

Eclipse. Eclipse Neon 3 Packages, 2017. Disponível em: <http://www.eclipse.org/downloads/packages/>. Acesso em: 02 Junho 2017.

Elsevier B.V. Corporate security solutions for BYOD: A novel user-centric and self-adaptive system, 2015.

Esler, Joel. GUIs for Snort, 2011. Disponível em: <http://blog.snort.org/2011/01/guis-for-snort.html>. Acesso em 21 Maio 2017

Ferril, Paul. SOTI MobiControl Review, 2016. Disponível em: <http://www.pcmag.com/review/345125/soti-mobicontrol>. Acesso em: 21 Maio 2017.

Figueiredo, Eduardo. Requisitos Funcionais e Requisitos Não Funcionais, 2016. p. 1-14. Disponível em: <http://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/req-funcional-rnf_v01.pdf>. Acesso em: 05 Junho 2017.

Galvão, Ricardo. Redes de Computadores, Camada de Rede (Endereçamento IP), 2016. p. 1-49. Disponível em: <https://docente.ifrn.edu.br/ricardogalvao/disciplinas/2016.1/tec.sis.1m-redes-de-computadores/.-slides-camada-de-rede-tcp-ip-enderecamento-ip>. Acesso em: 22 Maio 2017.

Gehardt, Tatiana., Silveira, Denise. Métodos de Pesquisa. p. 1-120, 2009. Disponível em: <http://www.ufrgs.br/cursopgdr/downloadsSerie/derad005.pdf >. Acesso em: 15 Maio 2017.

Gismo Freware. Cryptomator Review, 2017. Disponível em: <https://www.techsupportalert.com/content/cryptomator.htm-0 >. Acesso em: 22 Maio 2017.

Graças, Sergio. JNETMAP - Monitoramento Gráfico De Rede, 2013. Disponível em: <http://tecnicolinux.blogspot.com.br/2013/07/jnetmap-monitoramento-grafico-de-rede.html>. Acesso em: 16 Setembro 2017.

Gulzar, Nadir. Fast Track to Struts: What id Does and How. p. 1-29, 2002. Disponível em: <http://media.techtarget.com/tss/static/articles/content/StrutsFastTrack/StrutsFastTrack.pdf>. Acesso em: 13 Junho 2017.

Haataja, K., Hyppönen, K., Pasanen, S., Toivanen, P. Bluetooth Security Attacks: Comparative Analysis, Attacks and Countermeasures, p. 1-92, 2013.

InfoWorld. 7 killer open source monitoring tools, 2014. Disponível em: <http://www.infoworld.com/article/2683857/network-monitoring/article.html#slide8>. Acesso em: 21 Maio 2017.

Isaca. Cybersecurity Fundamentals Glossary, p. 1-35, 2016. Disponível em: <http://www.isaca.org/knowledge-center/documents/glossary/cybersecurity_fundamentals_glossary.pdf>. Acesso em: 13 Abril 2017.

Isaca. Isaca Journal Volume 1, p. 1-6, 2012. Disponível em: <https://www.isaca.org/Journal/archives/2012/Volume-1/Documents/12v1-Database-Backup.pdf >. Acesso em: 16 Maio 2017.

Jakobsson, M., Wetzel, S. Security Weaknessess in Bluetooth, 2001. Disponível em: <http://www.csd.uoc.gr/~hy544/mini_projects/Project2/Security%20Weaknesses%20in%20Bluetooth.pdf>. Acesso em: 04 Abril 2017.

JGRCS. Bring Your Own Device (BYOD): Security Risks and Mitigating Strategies, 2010. Disponível em: < http://www.jgrcs.info/index.php/jgrcs/article/view/654>. Acesso em: 10 Abril 2017.

Kali.org., Kali Tools. Aircrack-ng, 2014. Disponivel em: <http://tools.kali.org/wireless-attacks/aircrack-ng>. Acesso em: 22 Maio 2017.

Kaspersky Blog. O que são Rootkits e como Enfrentá-los, 2013. Disponível em: <https://blog.kaspersky.com.br/o-que-sao-rootkits-e-como-enfrenta-los/769/>. Acesso em: 22 Maio 2017.

Kirk, Jeremy. Instagram, Grindr, and more popular Android apps put user privacy at risk, researcher says, 2014. Disponível em: <https://www.pcworld.com/article/2603900/popular-

android-apps-fail-basic-security-tests-putting-privacy-at-risk.html>. Acesso em: 15 Setembro 2017.

Martinez, Erick. Snort com MySQL e AcidBase, 2015. Disponível em: <https://www.mundotibrasil.com.br/snort-com-mysql-e-acidbase/#more-3802>. Acesso em: 21 Maio 2017.

Mengue, Fábio. Curso de Java Básico, p. 1-35, 2002. Disponível em: <ftp://ftp.unicamp.br/pub/apoio/treinamentos/linguagens/java_basico.pdf>. Acesso em: 30 Maio 2017.

Mohamed, Ahmed. Password Cracking Using Cain & Abel, 2013. Disponível em: <http://resources.infosecinstitute.com/password-cracking-using-cain-abel/#gref>. Acesso em: 21 Maio 2017.

Montoro, Massimiliano. Cain & Abel, 2014. Disponível em: < http://www.oxid.it/cain.html>. Acesso em: 21 Maio 2017.

Morrow, Bill. BYOD security challenges: control and protect your most sensitive data, 2012. Disponível em: <http://ac.els-cdn.com/S1353485812701113/1-s2.0-S1353485812701113-main.pdf?_tid=c89b8a76-13b2-11e7-b676-00000aacb362&acdnat=1490704605_ab0fb0842c0b93960949633cf8afd270>. Acesso em: 28 Março 2017.

Netresec. NetworkMiner, 2017. Disponível em: <http://www.netresec.com/?page=NetworkMiner>. Acesso em: 15 Setembro 2017.

Null Byte. Getting Started with the Aircrack-Ng Suite of Wi-Fi Hacking Tools, 2013. Disponível em: < https://null-byte.wonderhowto.com/how-to/hack-wi-fi-getting-started-with-aircrack-ng-suite-wi-fi-hacking-tools-0147893/>. Acesso em: 29 Maio 2017.

Olavrsud, Thor. IT and Employees See BYOD Security (Much) Differently, 2012. Disponível em: <http://www.cio.com/article/2390217/mobile/it-and-employees-see-byod-security--much--differently.html>. Acesso em: 27 Março 2017.

Open Visual Traceroute. Disponível em: <http://visualtraceroute.net/>. Acesso em: 21 Maio 2017.

Oracle. The Java Programming Language and the Java Platform, 2017. Disponível em: <http://www.oracle.com/technetwork/topics/newtojava/downloads/index.html>. Acesso em: 30 Maio 2017.

Oracle. Windows System Requirements for JDK and JRE, 2016. Disponível em: <https://docs.oracle.com/javase/8/docs/technotes/guides/install/windows_system_requirements.html#BABIFDGD>. Acesso em: 02 Junho 2017.

Radu, Micu. NetCalulator; A simple net calculator for use in networking math calculations, 2005. Disponível em: <https://www.codeproject.com/Articles/11063/NetCalculator>. Acesso em: 22 Maio 2017.

Rakudave. jNetMap Network Monitoring Tool, 2017. Disponível em: <http://www.rakudave.ch/jnetmap/?file=introduction>. Acesso em: 16 Setembro 2017.

Richard Stevens, W., Fenner, B., Ruddof, A.M. UNIX Network Programming, Volume 1,p 1-947, 2003. Disponível em: <https://books.google.com.br/books?id=ptSC4LpwGA0C&pg=PA52&dq=socket+pair+tuple&redir_esc=y#v=onepage&q=socket%20pair%20tuple&f=false>. Acesso em: 22 Outubro 2017.

Rubens, Paul. Secure Your WLAN With Aircrack-ng, 2007. Disponível em: <http://www.enterprisenetworkingplanet.com/netsecur/article.php/3718671/Secure-Your-WLAN-With-Aircrackng.htm>. Acesso em: 22 Maio 2017.

San José, Calif. Cisco Study: IT Saying Yes To BYOD, 2012. Disponível em: <https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=854754>. Acesso em: 27 Março 2017.

Sommerville, Ian. Engenharia de Software. 9ª Edição, 2011. ed. Pearson Education – BR.

Osei, Ezer, Boaten, Francis. International Journal in IT and Engineering Vol. 04: Bring-Your-Own-Device (BYOD): An Evaluation of Associated Risks to Corporate Information Security, 2016. Disponível em: <https://arxiv.org/ftp/arxiv/papers/1609/1609.01821.pdf> . Acesso em: 11 Abril 2017.

Montgomery, Eric. Password Strength Meter, 2010. Disponível em: <https://github.com/ericmdw/java-pwdstrength>. Acesso em: 22 Maio 2017.

Nasa. Nasa Software 2017–2018 Catalog, p 1-154, 2017. Disponível em: <https://software.nasa.gov/NASA_Software_Catalog_2017-18.pdf>. Acesso em: 16 Maio 2017.

Ntop. Disponível em: < http://www.ntop.org/>. Acesso em: 21 Maio 2017.

Phatak, Prashant. Top 10 Security Assessment Tools, 2012. Disponível em: <http://opensourceforu.com/2012/02/top-10-security-assessment-tools/ >. Acesso em: 16 Maio 2017.

Pinheiro, Ricardo. Ntopng – Conheça a nova versão do ntop, 2014. Disponível em: <https://www.mundotibrasil.com.br/ntopng-conheca-a-nova-versao-ntop/>. Acesso em: 21 Maio 2017.

PwC., Info Security Europe. 2015 Information SecurityBreaches Survey. p. 1-52, 2015. Disponível em: < https://www.pwc.co.uk/assets/pdf/2015-isbs-technical-report-blue-03.pdf >. Acesso em: 16 Maio 2017.

Sectools.org. Disponível em: < http://sectools.org/tool/cain/>. Acesso em: 21 Maio 2017.

Skoudis, Ed. Top 5 Hacker Tools: Google hacker, password cracker, WLAN detector, 2013. Disponível em: < http://searchsecurity.techtarget.com/magazineContent/Top-5-Hacker-Tools-Google-hacker-password-cracker-WLAN-detector>. Acesso em: 21 Maio 2017.

Snort. Disponível em: <https://www.snort.org/>. Acesso em: 21 Maio 2017.

Soti. MobiControl. Disponível em: < https://www.soti.net/>. Acesso em: 21 Maio 2017.

Sommerfeld, Rafael. Como sobreviver ao paradoxo da Shadow IT x TI Convencional, 2015. Disponível em: <http://computerworld.com.br/como-sobreviver-ao-paradoxo-da-shadow-it-x-ti-convencional> Acesso em: 27 Março 2017.

Taurion, Cezar. Cloud Computing: computação em nuvem: transformando o mundo da tecnologia da informação, 2009. Editora Brasport: Rio de Janeiro, Brasil.

Technology Strategy Board - IoT Special Interest Group, 2013. Internet of Things (IoT) and Machine to Machine Communications (M2M) Challenges and opportunities: Final paper May 2013. Disponível em: <https://connect.innovateuk.org/documents/3077922/3726367/IoT+Challenges,%20final+pap

er,%20April+2013.pdf/38cc8448-6f8f-4f54-b8fd-3babed877d1a>.
Acesso em: 09 Março 2017.

Thomson, G. BYOD: Enabling the chaos. Network Security. 2012. Disponível em: <http://ac.els-cdn.com/S1353485812700132/1-s2.0-S1353485812700132-main.pdf?_tid=92dc444c-1930-11e7-b49f-00000aab0f02&acdnat=1491308387_1def13472b9f5763bbf3d839be478bec>. Acesso em: 04 Abril de 2017.

Universal Password Manager. Disponível em: <http://upm.sourceforge.net/index.html>. Acesso em: 22 Maio 2017.

Zanetti, Gabriel. IP Monitor. 2017. Disponível em: <https://github.com/pupi1985/IPMonitor>. Acesso em 22 Outubro 2017.

ANEXO A

Descrição dos casos de uso da Figura 17:

Caso de Uso 01: Começar Captura de Pacotes de Rede

Descrição: O usuário deverá selecionar qual interface de rede deseja utilizar e após clicar no botão de inicio da captura.

Atores envolvidos: Usuário do sistema.

Pré-condição: O usuário deverá ter selecionado qual adaptador de rede será escaneado.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *NetworkMiner*.
2. O usuário deve escolher qual adaptador de rede deseja escanear.
3. O usuário deve clicar no botão "*Start*".
4. A ferramenta executa a captura de pacotes de rede.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não executa a captura.

Caso de Uso 02: Parar Captura de Pacotes de Rede

Descrição: O usuário deverá finalizar a execução da ferramenta.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter começado uma captura de pacotes de rede.

Cenário Principal de Sucesso:

1. Após a listagem dos resultados obtidos na interface principal do *NetworkMiner*, o usuário deve clicar no botão "*Stop*".
2. A ferramenta finalizará o processo de escaneamento.

Cenário Secundário de Falha:

A ferramenta retorna a mensagem de erro e não finaliza a captura.

Caso de Uso 03: Filtrar Resultados da Captura

Descrição: O usuário deverá selecionar um endereço IP para filtrar os resultados exibidos.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa enviar um endereço IP do *jNetMap* para o *NetworkMiner* e o filtro precisa estar ligado.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *jNetMap*.
2. O usuário deve clicar com o botão direito sobre um dispositivo mapeado e então clicar no botão “Filtrar IP no *NetoworkMiner*”.
3. O usuário deve navegar até a ferramenta *NetworkMiner*.
4. O usuário deverá iniciar uma nova captura de pacotes manualmente.
5. O *NetworkMiner* mostrará apenas os pacotes de rede que contenham alguma relação com o endereço IP selecionado.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não filtra os resultados.

Caso de Uso 04: Criar Mapa da Topologia de Rede

Descrição: O usuário deverá criar um mapa para representar graficamente os dispositivos de rede.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa estar com a ferramenta *jNetMap* aberta.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *jNetMap*.
2. O usuário deve clicar no botão “*New Map*”.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não cria o mapa.

Caso de Uso 05: Adicionar Dispositivos ao Mapa

Descrição: O usuário deverá mapear os dispositivos que compõem a rede corporativa.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa estar com um mapa aberto.

Cenário Principal de Sucesso:

1. O usuário poderá adicionar os dispositivos manualmente clicando com o botão direito do mouse em uma área em branco do mapa.
2. O usuário deverá informar as características do dispositivo.
3. O usuário deverá clicar em “Ok” para adicionar o dispositivo ao mapa.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não adiciona o dispositivo.

Caso de Uso 06: Excluir Dispositivos do Mapa

Descrição: O usuário deverá excluir os dispositivos que compõem a rede corporativa.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter um mapa com dispositivos mapeados.

Cenário Principal de Sucesso:

1. O usuário deverá clicar com o botão direito do mouse sobre um dispositivo já mapeado e clicar no botão "*Delete*".

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não exclui o dispositivo.

Caso de Uso 07: Escanear Dispositivos Conectados na Rede

Descrição: O usuário deverá realizar uma varredura na rede detectando os dispositivos que estão conectados.

Pré-condição: Usuário precisa ter um mapa aberto.

Cenário Principal de Sucesso:

1. O usuário deverá navegar até o menu "*Tools*" e clicar no botão "*Network Scanner*".
2. O usuário deverá informar a faixa de IP e a máscara de rede e após clicar em "*Scan*".
3. Depois de finalizado, o usuário deverá selecionar quais dos dispositivos detectados ele deseja adicionar ao seu mapa e clicar em "*Add*".

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e realiza o escaneamento.

Caso de Uso 08: Escanear Portas em Uso do Dispositivo

Descrição: O usuário deverá escanear quais portas estão sendo utilizadas por um determinado dispositivo na rede.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter este dispositivo mapeado.

Cenário Principal de Sucesso:

1. O usuário deve clicar com o botão direito do mouse sobre o dispositivo e então clicar no botão "*Port Scan*".

2. O usuário deve informar se deseja fazer um escaneamento completo ou apenas de determinadas portas e então clicar em "*Scan*".

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não faz o escaneamento de portas.

Caso de Uso 09: Salvar e Exportar Mapa de Topologia de Rede

Descrição: O usuário deverá salvar o mapa que criou utilizando o *jNetMap*.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter um mapa aberto.

Cenário Principal de Sucesso:

1. O usuário deve clicar no ícone de salvar no menu superior ou navegar até o menu "Arquivo" e clicar no botão "Salvar".
2. O usuário deve informar qual o diretória de sua preferência e clicar em "Salvar".
3. Para exportar o mapa, o usuário deverá escolher qual o formato de arquivo e então clicar em "Salvar".

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não faz o escaneamento de portas.

Caso de Uso 10: Criar uma Base de Dados Criptografada

Descrição: O usuário deverá definir as informações básicas para criar uma base de dados.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter um diretório para salvar a base de dados.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Universal Password Strength Meter*.
2. O usuário deve definir qual diretório e chave de acesso utilizará.
3. O sistema salva as informações e confirma a criação através de uma mensagem.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não cria a base de dados.

Caso de Uso 11: Adicionar Informações de *Login*

Descrição: O usuário deverá inserir os dados de *login* para salvar na base de dados.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter usuários e senhas para salvar na base de dados criada previamente.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Universal Password Strength Meter*.
2. O usuário deve abrir a base de dados.
3. O usuário deve informar os dados de *login* que deseja armazenar.
4. O sistema salva as informações e confirma a inserção através de uma mensagem.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não armazena as informações.

Caso de Uso 12: Visualizar Informações de *Login*

Descrição: O usuário deverá inserir os dados de *login* para salvar na base de dados.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter usuários e senhas para salvar na base de dados criada previamente.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Universal Password Strength Meter*.
2. O usuário deve abrir a base de dados.
3. O usuário deve informar os dados de *login* que deseja armazenar.
4. O sistema salva as informações e confirma a inserção através de uma mensagem.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não armazena as informações.

Caso de Uso 13: Excluir Informações de *Login*

Descrição: O usuário deverá remover os dados de *login* salvos na base de dados.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa ter usuários e senhas cadastrados na base de dados previamente.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Universal Password Strength Meter*.
2. O usuário deve abrir a base de dados.
3. O usuário deve selecionar os dados de *login* que deseja remover.
4. O sistema remove as informações e confirma através de uma mensagem.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não remove as informações.

Caso de Uso 14: Excluir Base de Dados Criptografada

Descrição: O usuário deverá remover a base de dados de *login*.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa uma base de dados criada previamente.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Universal Password Strength Meter*.
2. Navegar até o menu "*Options*".
3. O usuário deve informar qual base de dados deseja remover.
4. O sistema remove as informações e confirma através de uma mensagem.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não remove as informações.

Caso de Uso 15: Testar Força de Senha

Descrição: O usuário deverá inserir uma senha para ser testada.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário deverá ter uma senha para testar.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *Password Strength Meter*.
2. O usuário deve informar a senha que deseja testar.
3. O usuário deve selecionar qual o nível de segurança que ele deseja testar e clicar em "Testar Senha".
4. O sistema realiza o teste e retorna o resultado através da interface gráfica.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não testa a senha.

Caso de Uso 16: Copiar Senha para Área de Transferência

Descrição: O usuário copiar a senha de forma rápida e prática para a Área de Transferência.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário deverá ter realizado o teste de senha.

Cenário Principal de Sucesso:

1. O usuário deve clicar no botão "Copiar senha para a Área de Transferência".

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e copia a senha.

Caso de Uso 17: Calcular Endereços de Rede e Subrede

Descrição: O usuário deverá definir qual classe de IP deseja calcular.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa informar uma faixa de IP e quantas subredes deseja calcular.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta *NetCalculator*.
2. O usuário deve informar os dados e selecionar qual classe de IP deseja calcular.
3. O sistema realiza os cálculos e mostra os resultados na tela.

Cenário Secundário de Falha:

1. A ferramenta retorna a mensagem de erro e não calcula as informações.

Caso de Uso 18: Monitorar Endereço IP Público

Descrição: O usuário poderá monitorar em tempo real o seu endereço de IP público.

Atores envolvidos: Usuário do sistema.

Pré-condição: Usuário precisa abrir a ferramenta IP Monitor.

Cenário Principal de Sucesso:

1. O usuário deve navegar até a ferramenta IP Monitor
2. O usuário deve selecionar o intervalo de tempo entre uma verificação e outra.
3. O usuário deverá personalizar qual o tipo de notificação, em caso de alteração no IP, deseja receber.
3. O usuário deverá clicar em "*Start*".

Cenário Secundário de Falha:

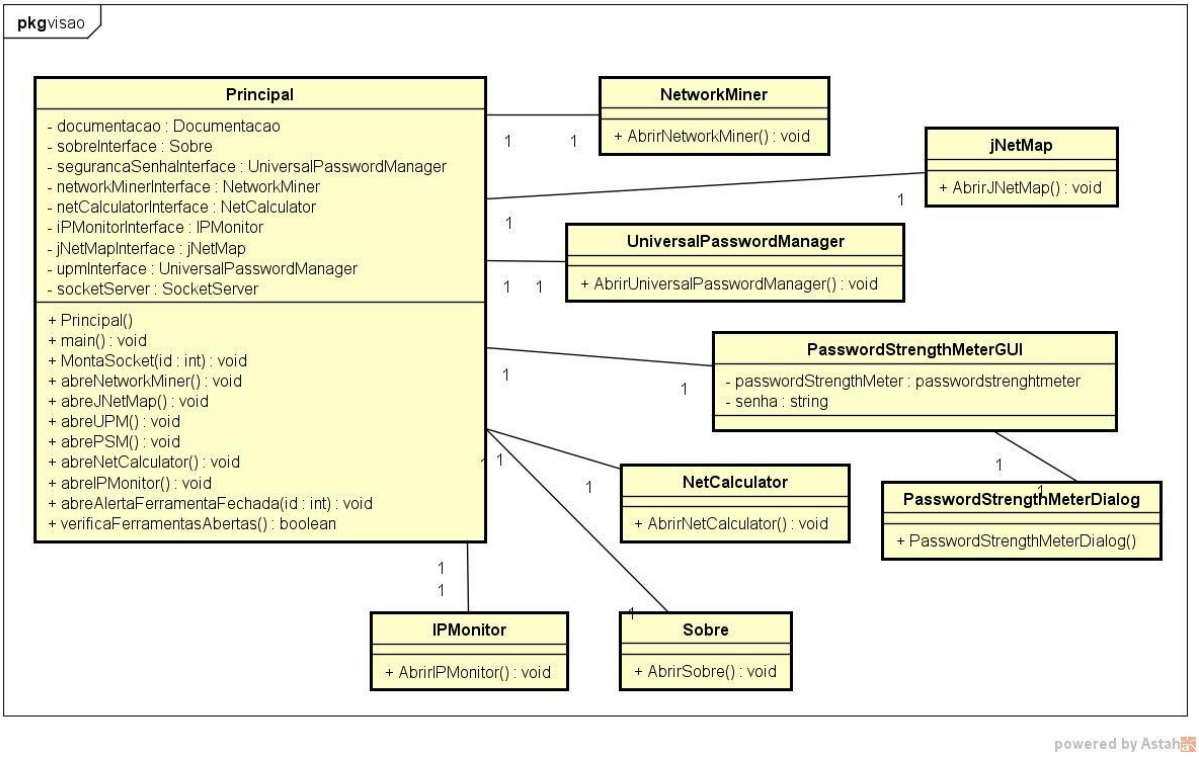
1. A ferramenta retorna a mensagem de erro e não começa o monitoramento.

ANEXO B

A Figura 68 representa o diagrama de classes da camada visão. Os objetivos específicos das classes mais importantes são:

- Principal: essa é a classe que contém a função *main*. Seu funcionamento consiste em exibir a interface principal de usuário e interpretar os eventos de cliques do usuário. Ela também é responsável por trocar dados com a camada controle, comunicando a abertura de uma nova ferramenta.
- Sobre: essa é a classe responsável por mostrar um pequeno diálogo mostrando ao usuário mais detalhes sobre o protótipo e as licenças de *software* livre.
- NetworkMiner: essa classe é responsável por mostrar uma pequena janela de carregamento enquanto o computador trabalha para abrir a ferramenta *NetworkMiner*.
- PasswordStrengthMeterGUI: essa classe é responsável por exibir uma interface gráfica que foi desenvolvida exclusivamente para a biblioteca *Password Strength Meter*. Como citado na seção 5.2.1, essa biblioteca foi importada duas vezes, sendo uma delas no projeto principal.
- As demais classes seguiram o modelo da classe NetworkMiner porém a janela de carregamento foi desativada por parâmetro pois há ferramentas que já possuem uma janela de carregamento própria ou inicializam de maneira rápida.

Figura 68 - Diagrama de Classes da Camada Visão.



Fonte: Próprio Autor, 2017.

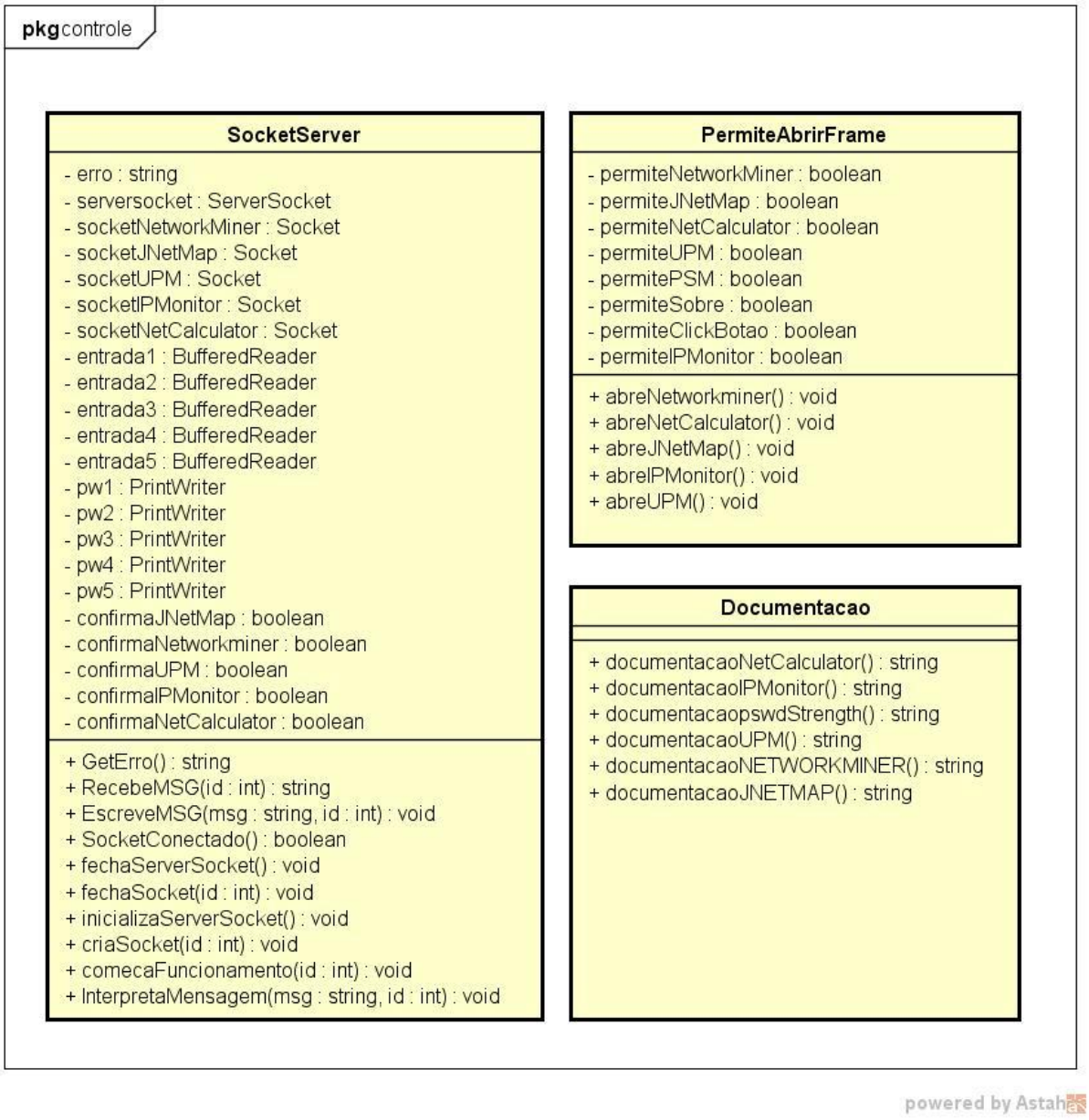
A Figura 69 representa o diagrama de classes da camada controle. Os objetivos específicos das classes mais importantes são:

- SocketServer: essa é a classe mais importante em todo o processo de integração das ferramentas. Essa classe é responsável por criar o *server socket*, estabelecer a conexão dos *clients sockets*, e por enviar mensagens e interpretar os dados recebidos. Desta forma é concretizada a comunicação entre os processos.

- PermiteAbrirFrame: essa classe é responsável pelos *flags* que garantem que uma ferramenta não seja inicializada mais de uma vez simultaneamente. Além disso, essa classe possui as funções que realizam as chamadas de inicialização das ferramentas.

- Documentacao: essa classe é responsável por detectar se o sistema possui um navegador de internet para abrir o site oficial de algumas ferramentas. Além disso, essa classe também detecta se existe um leitor de PDF instalado para abrir a documentação oficial das demais ferramentas.

Figura 69 - Diagrama de Classes da Camada Controle.



Fonte: Próprio Autor, 2017.

As Figuras 70, 71, 72 e 73 representam o diagrama de classes da camada modelo. Os objetivos específicos das classes mais importantes são:

- SocketClient (NetworkMiner): essa classe é responsável por se conectar com o servidor criado na camada controle, por enviar e receber informações e por interpretar os dados recebidos.
- Filtro (NetworkMiner): essa classe auxilia a filtragem dos resultados exibidos na interface principal. Além disso, ela é responsável por controlar o tempo de

escaneamento automático, ou seja, por quanto tempo a ferramenta vai capturar pacotes de rede que contenham o IP filtrado.

- SocketClient (jNetMap): essa classe é responsável por se conectar com o servidor criado na camada controle, por enviar e receber informações e por interpretar os dados recebidos.

- FilaDevices (jNetMap): essa classe é responsável por armazenar em fila os dados recebidos do *NetworkMiner*. Esta classe possibilita que o usuário importe os dispositivos respeitando o modelo FIFO (*First in, First Out* – Primeiro a Entrar, Primeiro a Sair).

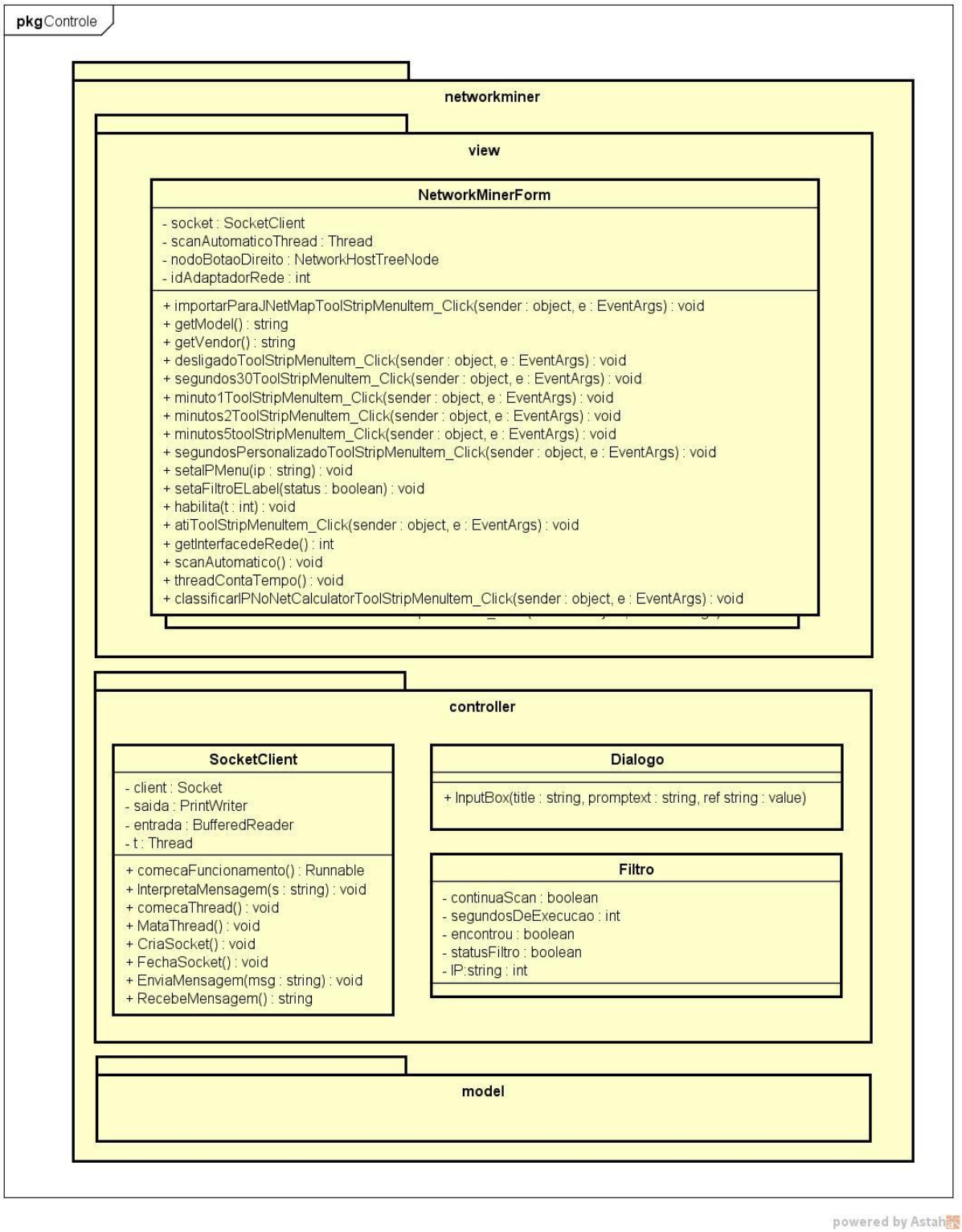
- SocketClient (IP Monitor): essa classe é responsável por se conectar com o servidor criado na camada controle, por enviar e receber informações e por interpretar os dados recebidos.

- SocketClient (NetCalculator): essa classe é responsável por se conectar com o servidor criado na camada controle, por enviar e receber informações e por interpretar os dados recebidos.

- SocketClient (Universal Password Manager): essa classe é responsável por se conectar com o servidor criado na camada controle, por enviar e receber informações e por interpretar os dados recebidos.

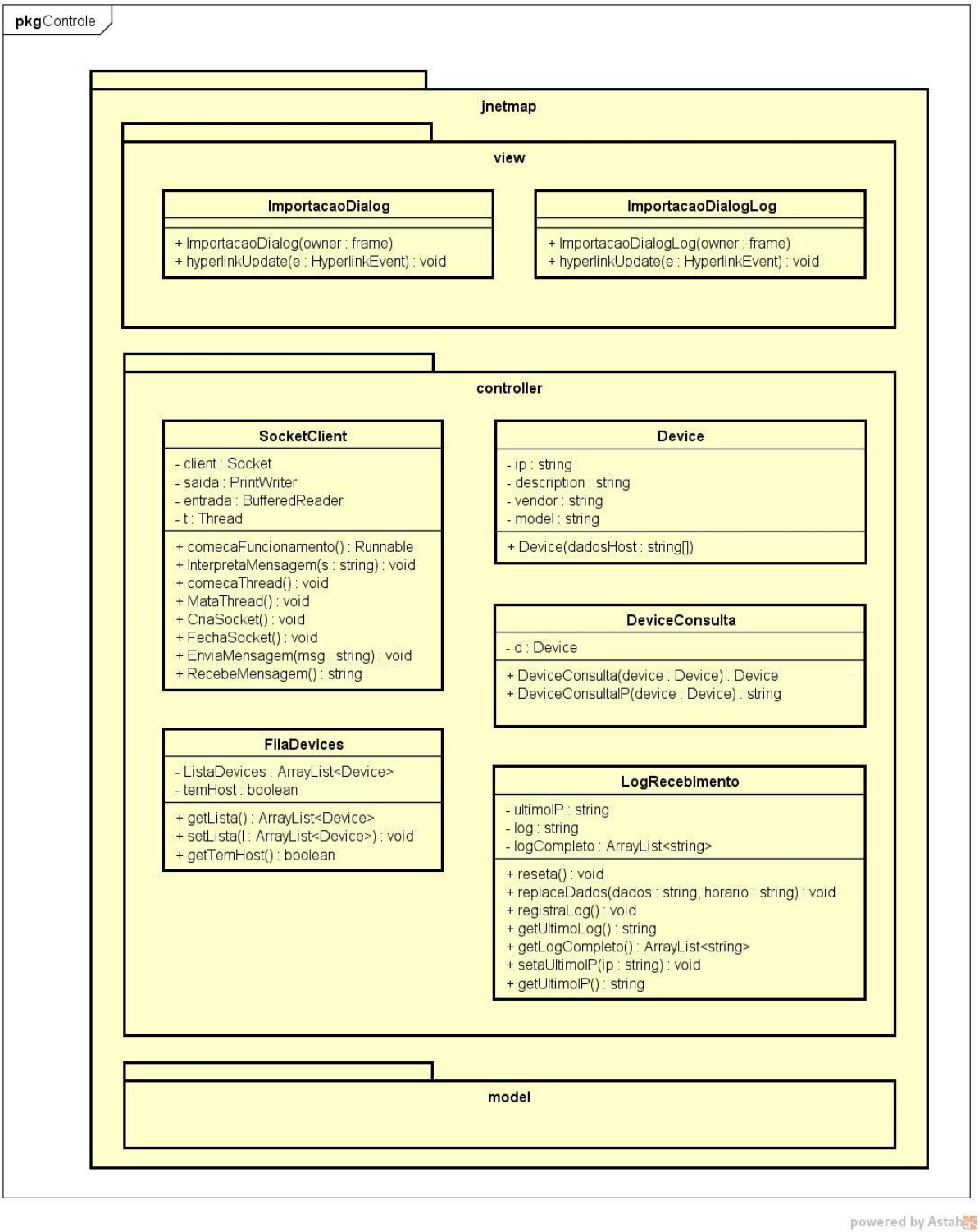
- PasswordStrengthMeterGUI (Universal Password Manager): essa classe é responsável por exibir uma interface gráfica que foi desenvolvida exclusivamente para a biblioteca *Password Strength Meter*. Como citado na seção 5.2.1, essa biblioteca foi importada duas vezes, sendo uma delas no projeto do *Universal Password Manager*.

Figura 70 - Diagrama de Classes do NetworkMiner.



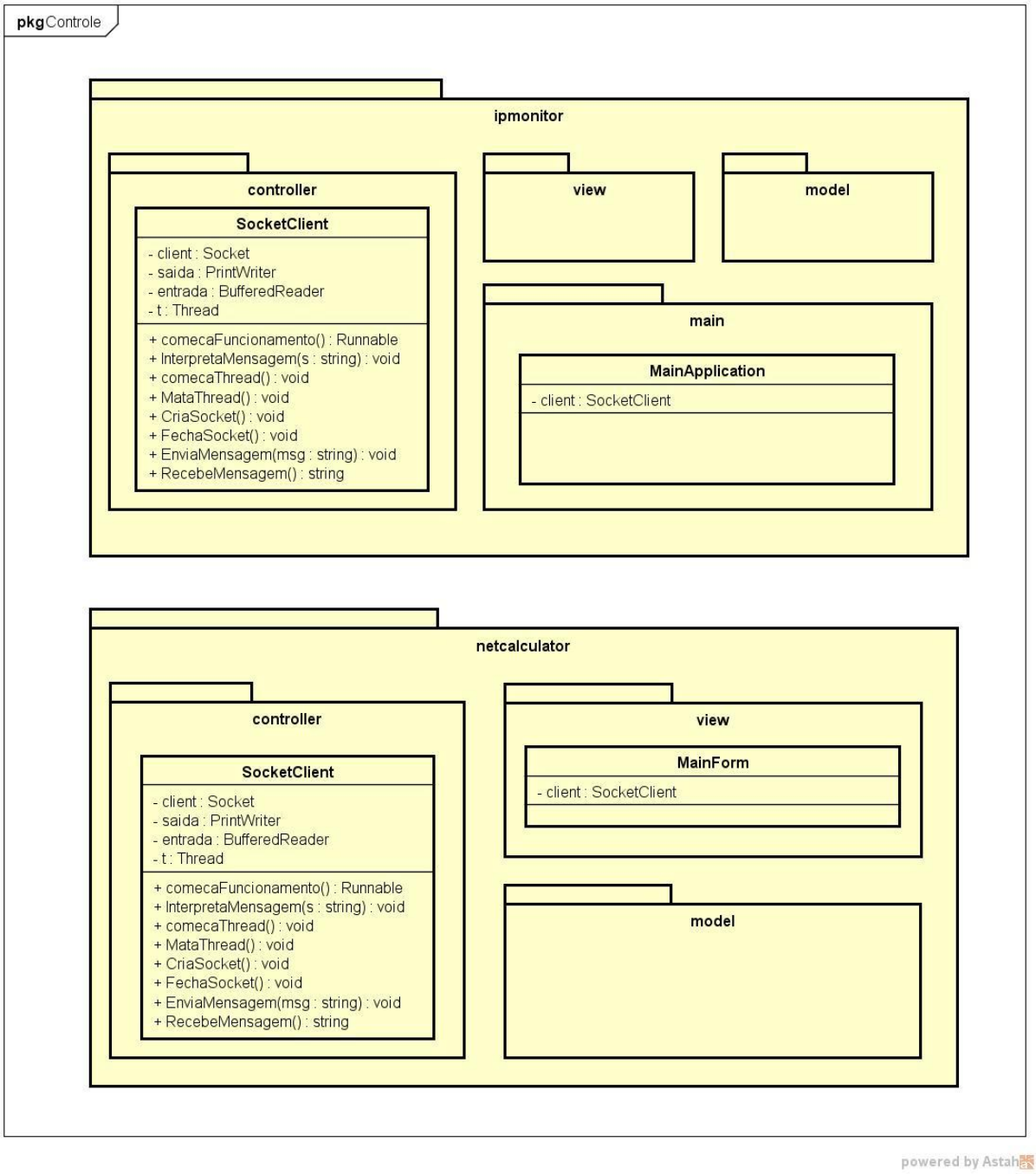
Fonte: Próprio Autor, 2017.

Figura 71 - Diagrama de Classes do jNetMap.



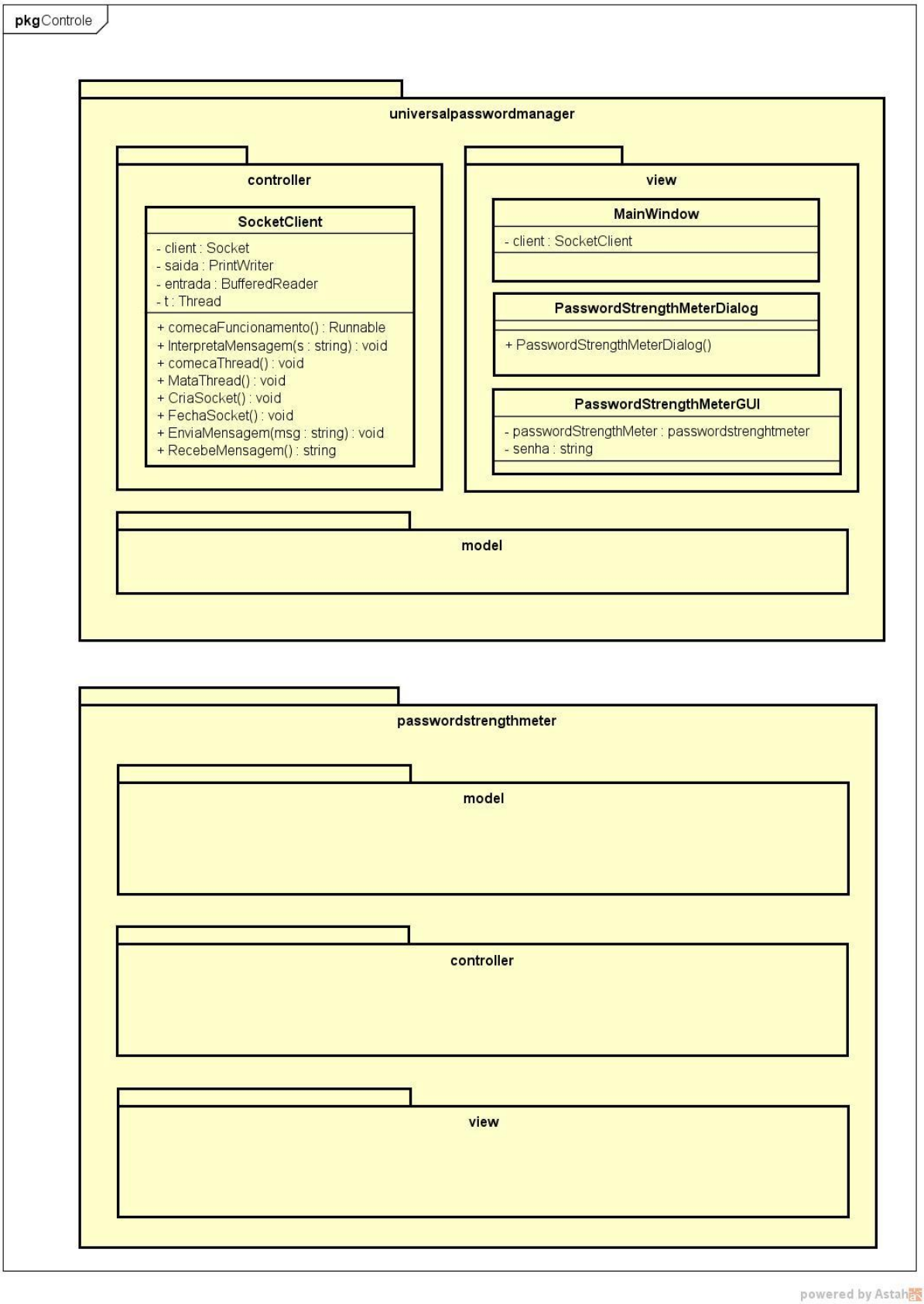
Fonte: Próprio Autor, 2017.

Figura 72 - Diagrama de Classes do IP Monitor e do NetCalculator.



Fonte: Próprio Autor, 2017.

Figura 73 - Diagrama de Classes do Universal Password Manager e do Password Strength Meter.



Fonte: Próprio Autor, 2017.

ANEXO C

BYOD MANAGER KIT
SUÍTE DE FERRAMENTAS OPEN-SOURCE PARA ADMINISTRAÇÃO E SEGURANÇA DE REDES BYOD

MANUAL DO USUÁRIO

Desenvolvido por: Vinícius Lahm Perini

Universidade de Caxias do Sul
Área do Conhecimento de Ciências Exatas e Engenharias
Bacharelado em Ciência da Computação
Caxias do Sul, 2017.

Sumário

O que é o BYOD Manager Kit?

O BYOD Manager Kit é um software de integração de ferramentas open-source para administração e segurança de redes BYOD que visa auxiliar as tarefas diárias do profissional de TI. Redes BYOD são redes corporativas nas quais os funcionários podem conectar dispositivos pessoais, como um notebook, por exemplo, para realizar tarefas profissionais.

O BYOD Manager Kit é uma suíte de ferramentas formada por um conjunto de vários outros softwares que foram alterados para funcionarem de maneira conjunta. Estes softwares foram divididos em categorias de acordo com a sua respectiva área de atuação.

Interface Principal

A interface principal do software é responsável por controlar e permitir o acesso às ferramentas integradas. O usuário[22] pode utilizar os menus ou os botões de atalho para abrir as ferramentas.

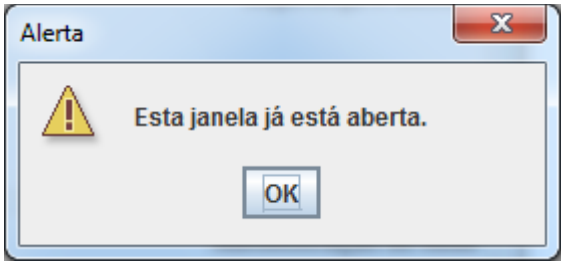
Interface Principal.

A interface é responsiva e funciona em diversas resoluções de tela. A ferramenta inicializa em um tamanho pré-fixado. Ao clicar na opção “Maximizar” a interface vai automaticamente se ajustar à resolução utilizada pelo usuário.

[22] Em todos os momentos em que este manual se refere ao termo “usuário” é com o intuito de identificar o usuário da ferramenta e não o usuário da rede corporativa. O usuário desta suíte possivelmente será um Administrador de Redes.

Detalhe do botão Maximizar.

Para garantir o correto funcionamento, o protótipo não permite que uma ferramenta seja aberta mais de uma vez, caso ela já esteja aberta.

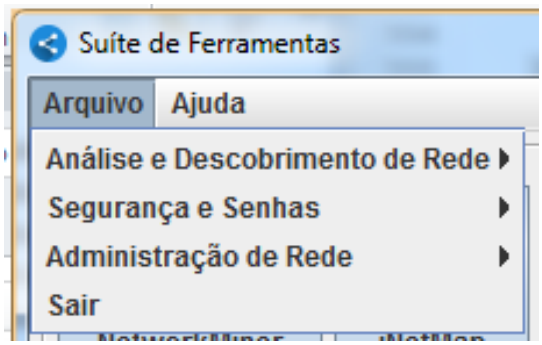


Caso o usuário tente abrir duas vezes a mesma ferramenta.

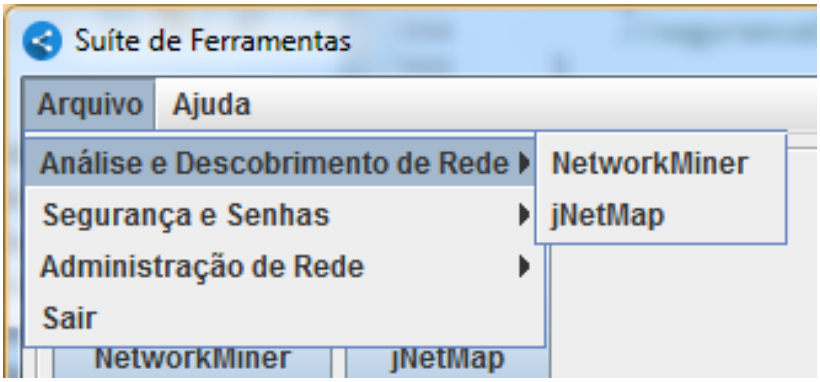
Além disso, o protótipo permite a inicialização de apenas uma ferramenta por vez. No momento em que o usuário clicar em um menu/botão, a ferramenta será carregada. Durante um curtíssimo intervalo de tempo os demais menus/botões ficarão desabilitados. Este processo garante o correto funcionamento das ferramentas.

Menu Arquivo

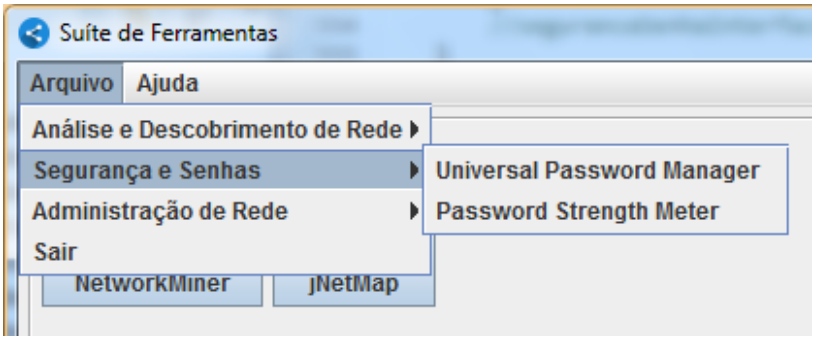
O menu Arquivo é responsável por disponibilizar o acesso às ferramentas que foram integradas de acordo com sua área de atuação. Na área “Análise e Descobrimento de Rede” é possível acessar as ferramentas: NetworkMiner e jNetMap. Na área “Segurança e Senhas” é possível abrir o Universal Password Manager e o Password Strength Meter. Na área “Administração de Rede” é possível abrir o IP Monitor e o NetCalculator. O menu Arquivo ainda possui a opção “Sair” que fecha o programa.



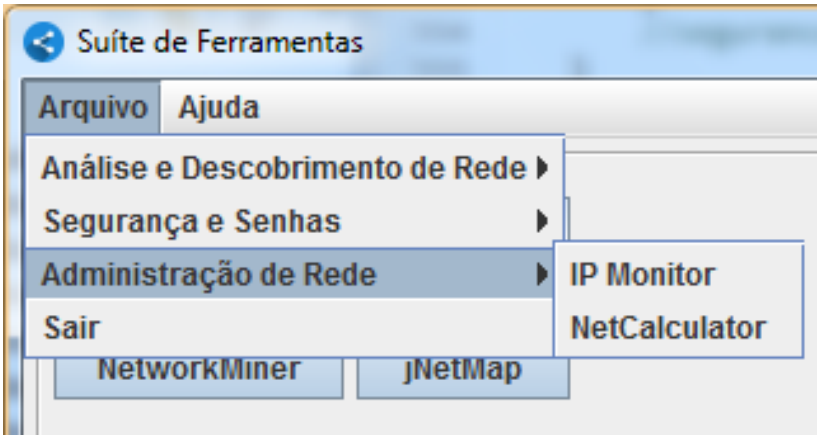
Menu Arquivo.



Menu Arquivo >> Análise e Descobrimento de Rede.



Menu Arquivo >> Segurança e Senhas.

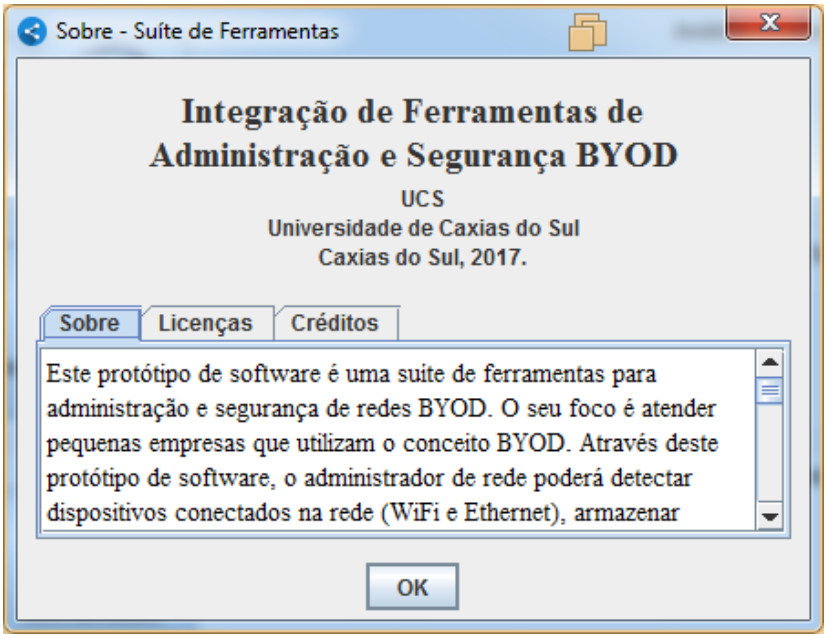


Menu Arquivo >> Administração de Rede.

Menu Ajuda

O menu Ajuda é responsável por permitir que o usuário acesse a documentação oficial de cada ferramenta com apenas um clique. A ferramenta NetworkMiner e a ferramenta jNetMap possuem manuais de usuário em arquivo PDF. As outras ferramentas disponibilizam sua documentação no site oficial. Além disso o menu Ajuda possui uma aba “Sobre” que abre um diálogo com informações adicionais sobre o projeto de integração.

Menu Ajuda >> Documentação.



Diálogo Sobre.

Botões de Atalho

Ainda na interface principal do BYOD Manager Kit, existem os botões de atalho. Esta é a forma mais rápida de acessar as ferramentas que foram integradas.

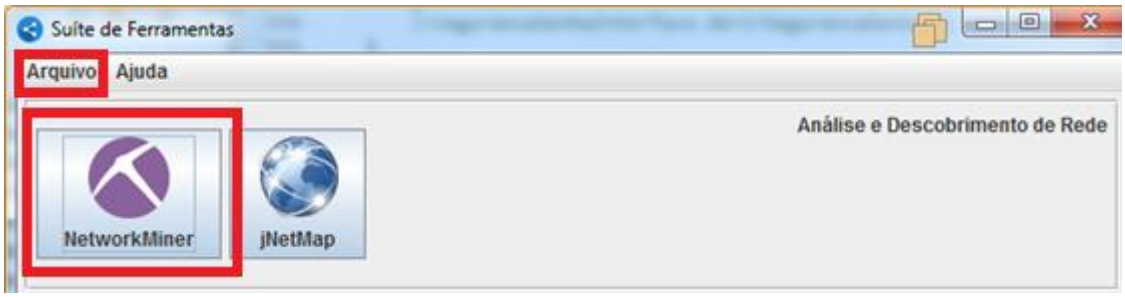
Botões de Atalho.

Análise e Descobrimento de Rede

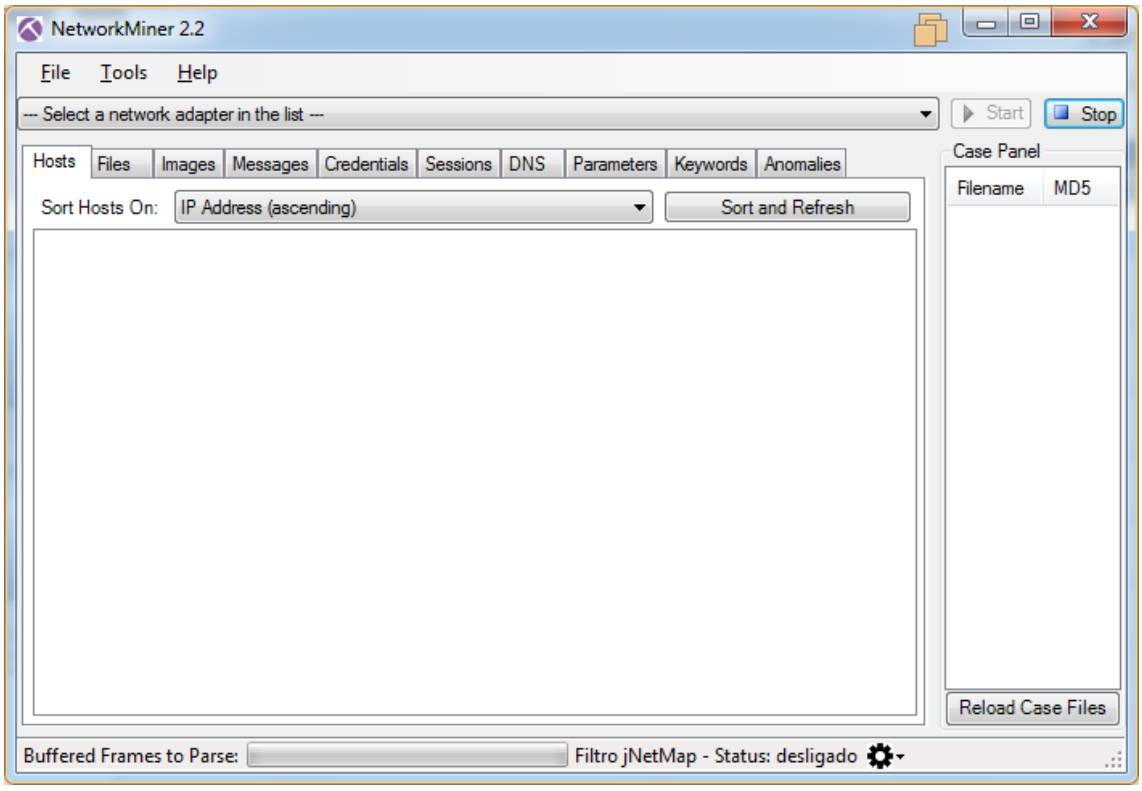
A categoria Análise e Descobrimento de Rede engloba duas ferramentas: o NetworkMiner e o jNetMap. O funcionamento delas é descrito a seguir. Após, as funcionalidades de integração são detalhadas.

NetworkMiner

O NetworkMiner é um sniffer de rede passivo, open-source e desenvolvido em C#. Ele pode ser acessado tanto pelo Menu Arquivo quanto pelo botão "NetworkMiner".



Em detalhe as formas de acessar o NetworkMiner.



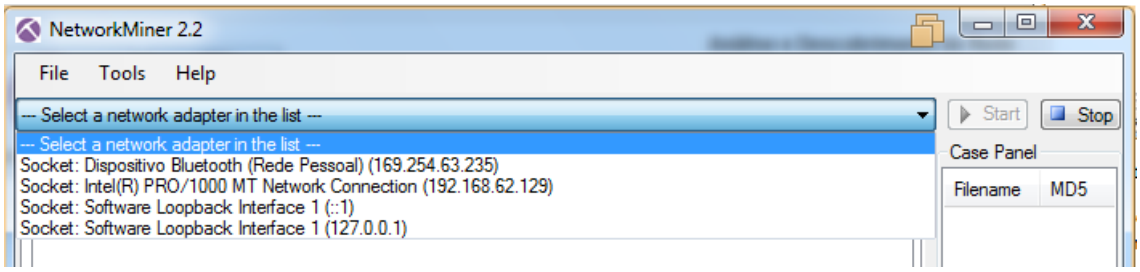
Interface principal do NetworkMiner integrado.

Para utilizar o NetworkMiner é necessário:

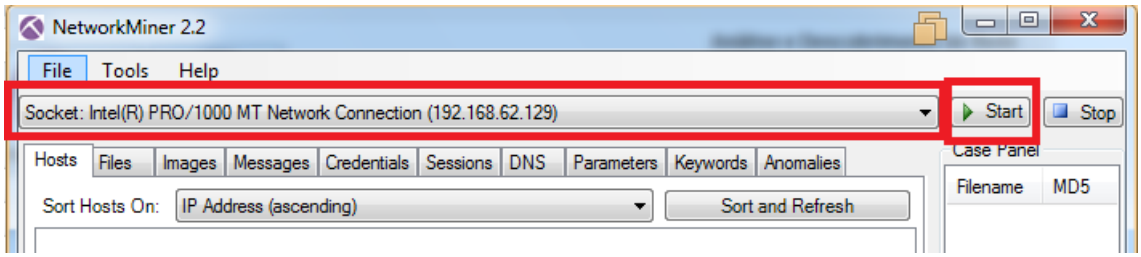
- Escolher um adaptador de rede dentre os disponíveis na lista (passo 1).
- Clicar no botão "Start" (passo 2).

- O software vai listar os resultados na tela, em suas diferentes abas.
- Caso haja o desejo de interromper o escaneamento de rede, clicar em “Stop” (passo 3).

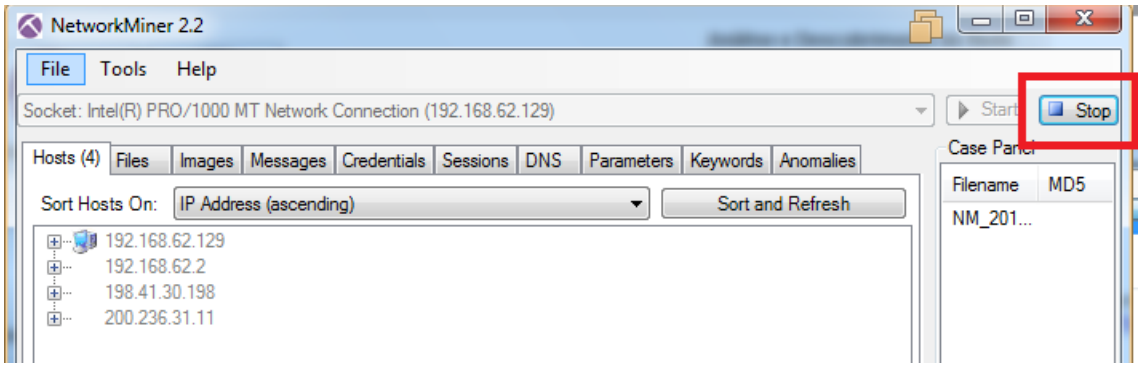
Maiores informações estão disponíveis na documentação oficial, que pode ser acessada no menu Ajuda na interface principal do BYOD Manager Kit.



Passo 1.



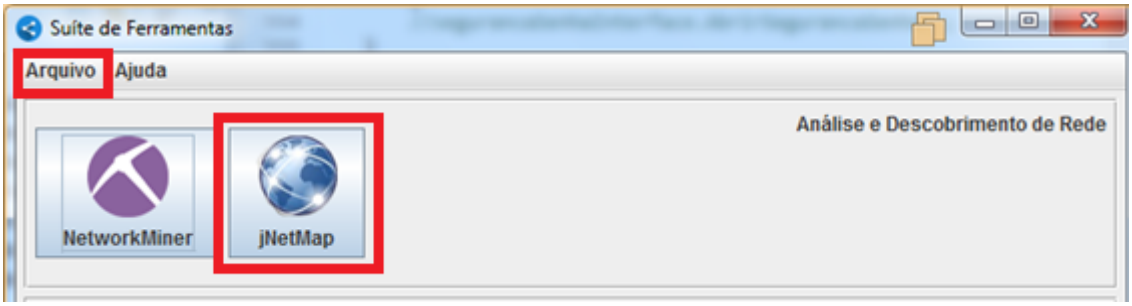
Passo 2.



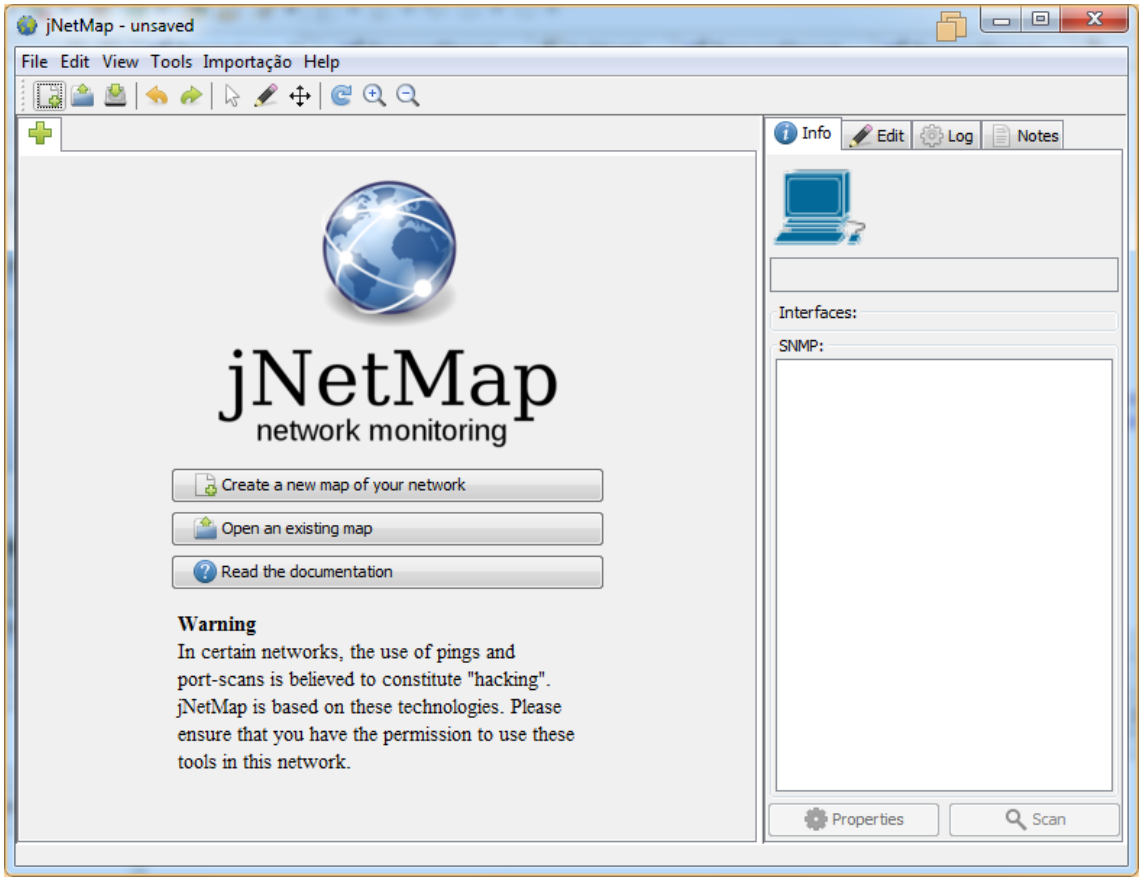
Passo 3.

jNetMap

O jNetMap é uma ferramenta gráfica para monitoramento e documentação de redes, open-source e desenvolvido em Java. Ela pode ser acessada tanto pelo Menu Arquivo quanto pelo botão “jNetMap”.



Em detalhe as formas de acessar o jNetMap.

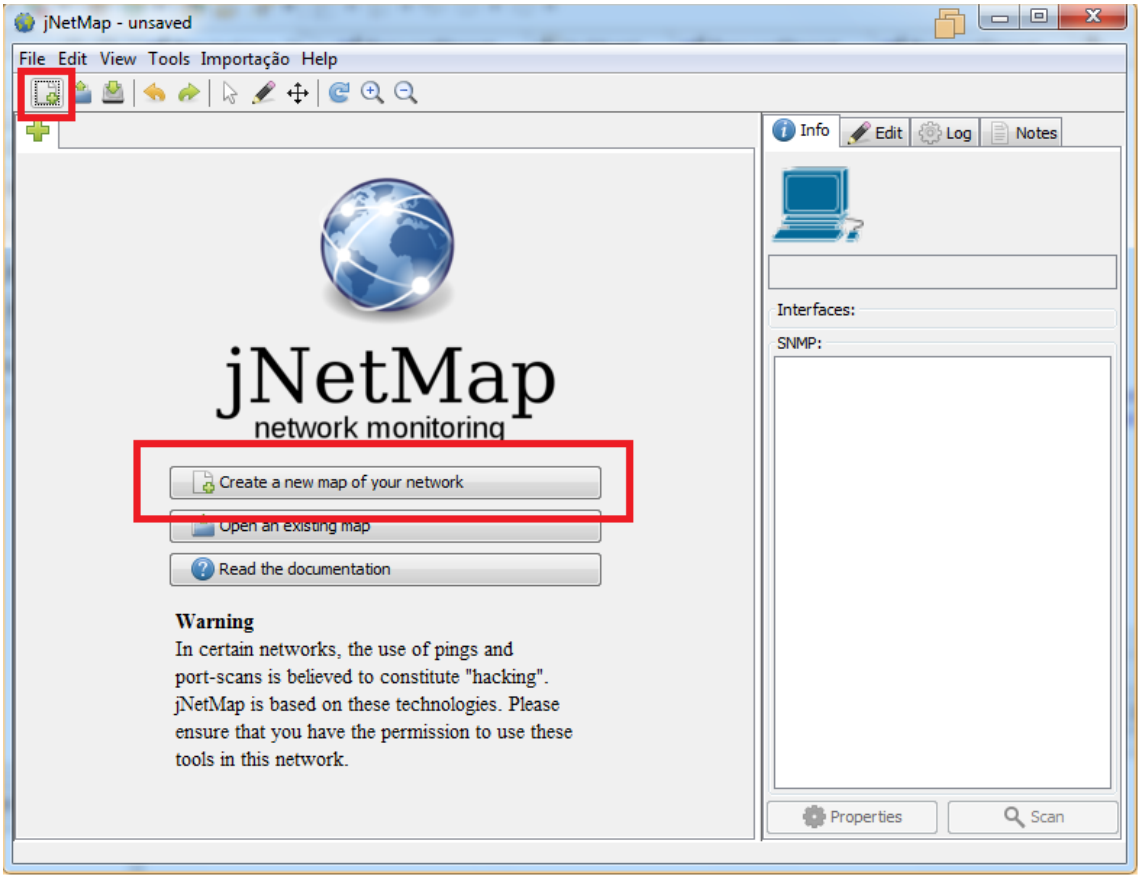


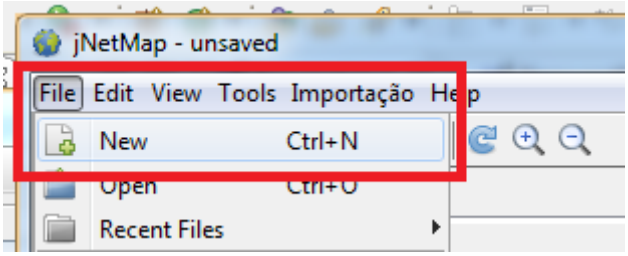
Interface principal do jNetMap integrado.

O jNetMap permite criar um novo mapa, abrir um mapa existente, adicionar manualmente dispositivos ou buscá-los na rede.

Para adicionar um novo mapa é necessário:

- Clicar no ícone novo mapa ou no botão “Create a new map of your network” ou através do menu “Arquivo” (passo 1).

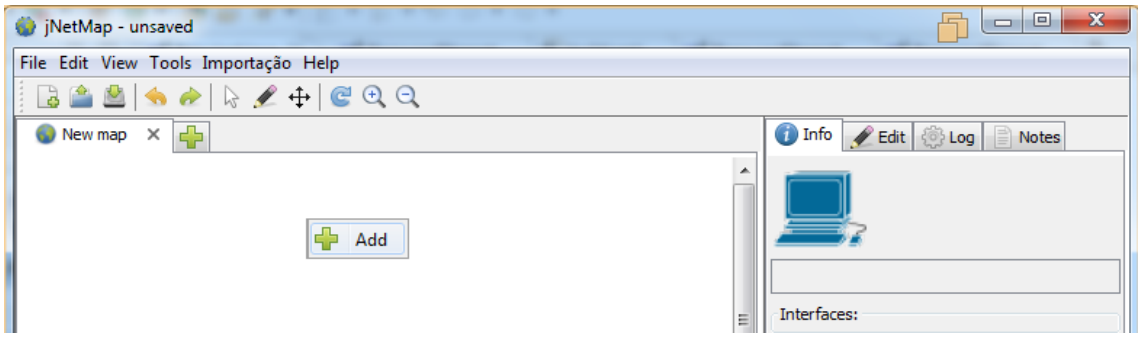




Como criar um novo mapa (passo 1).

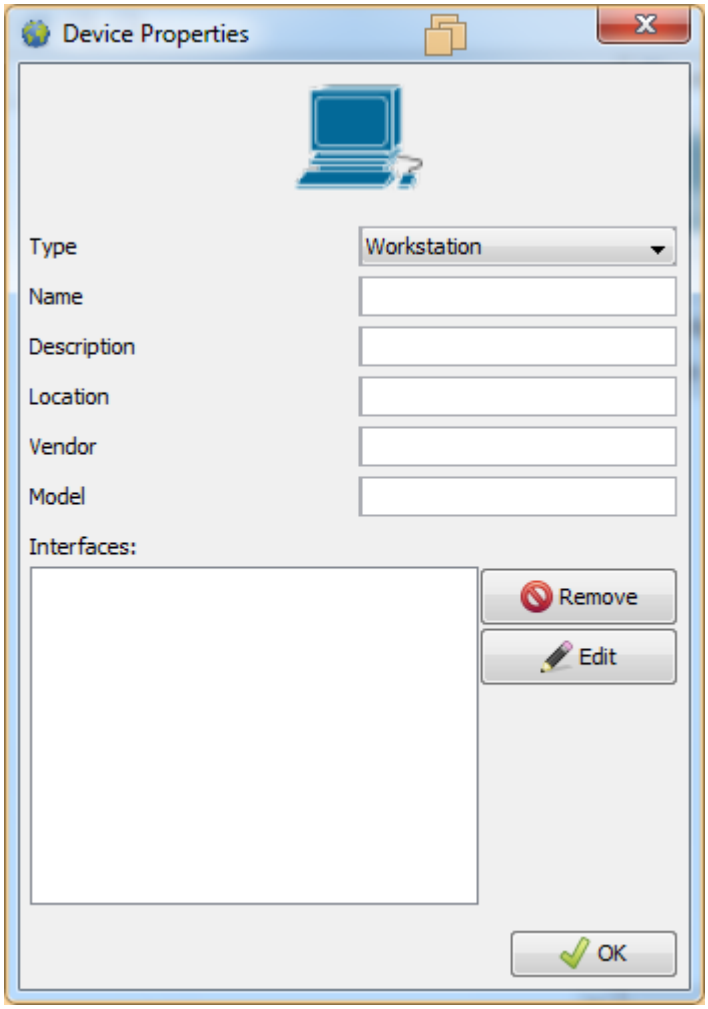
Para adicionar dispositivos manualmente:

- Clique com o botão direito do mouse em uma área em branco do mapa e então clicar no botão “Add” (passo 2).



Adicionar dispositivo (passo 2).

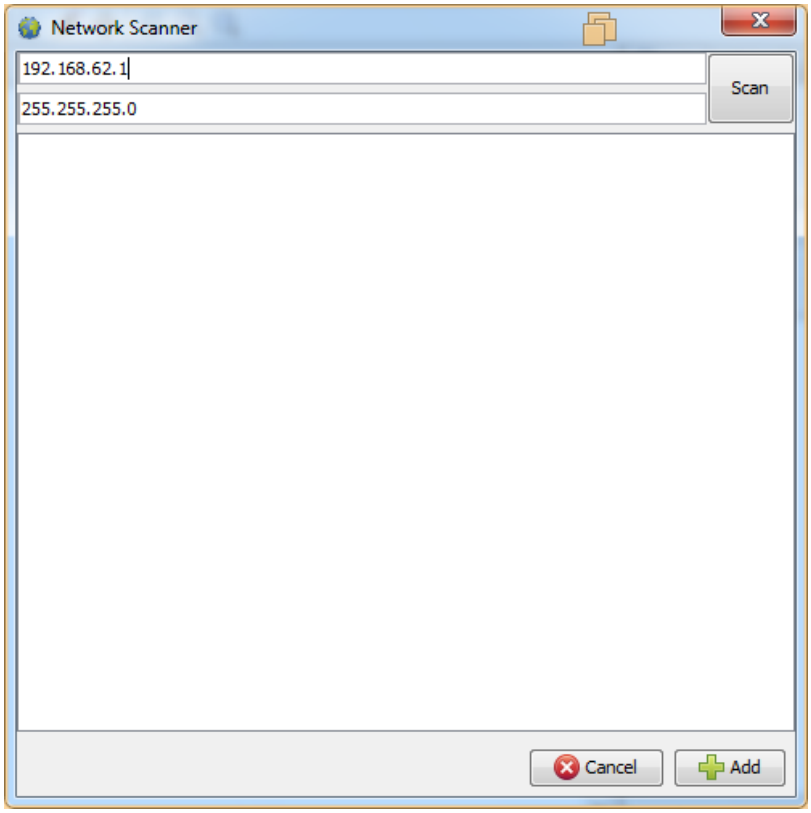
- Complete os campos de acordo com o dispositivo escolhido. Após clicar no botão “OK”.



Informando os dados do dispositivo (passo 2).

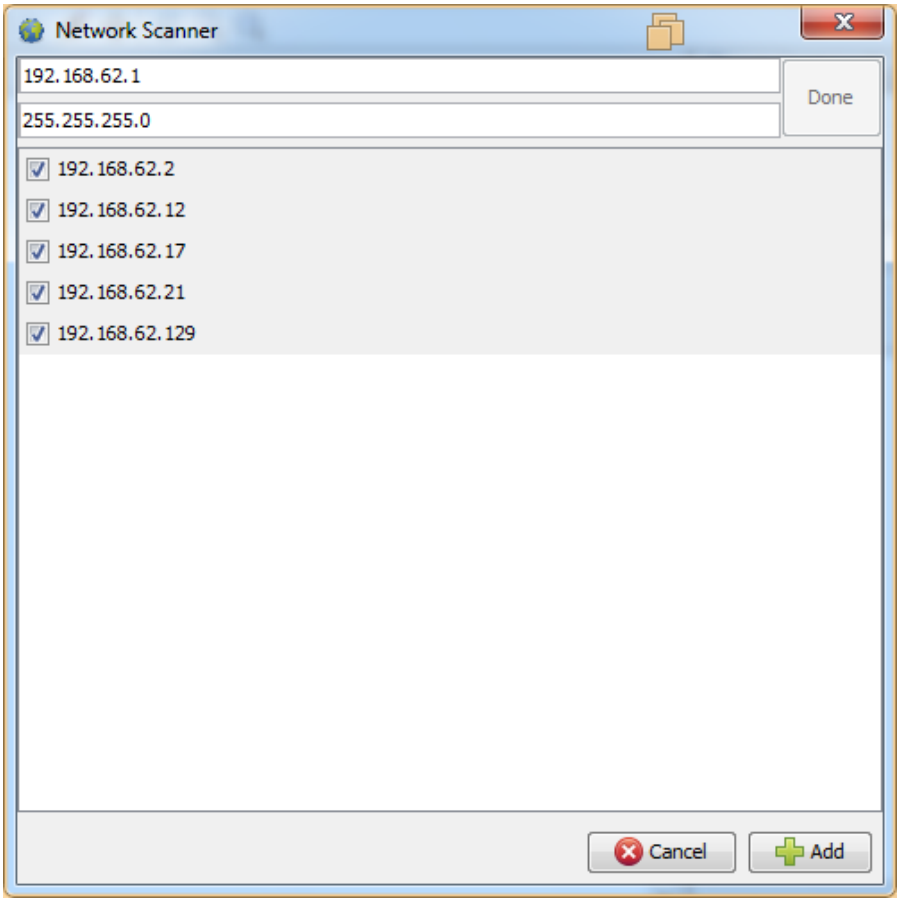
Para adicionar dispositivos de forma automática:

- Clique no menu “Tools” (Ferramentas) na interface principal do jNetMap.
- Clique no botão “Network Scanner”, que abrirá uma nova janela.
- Informe qual a faixa de IP deseja escanear e qual a máscara de rede.

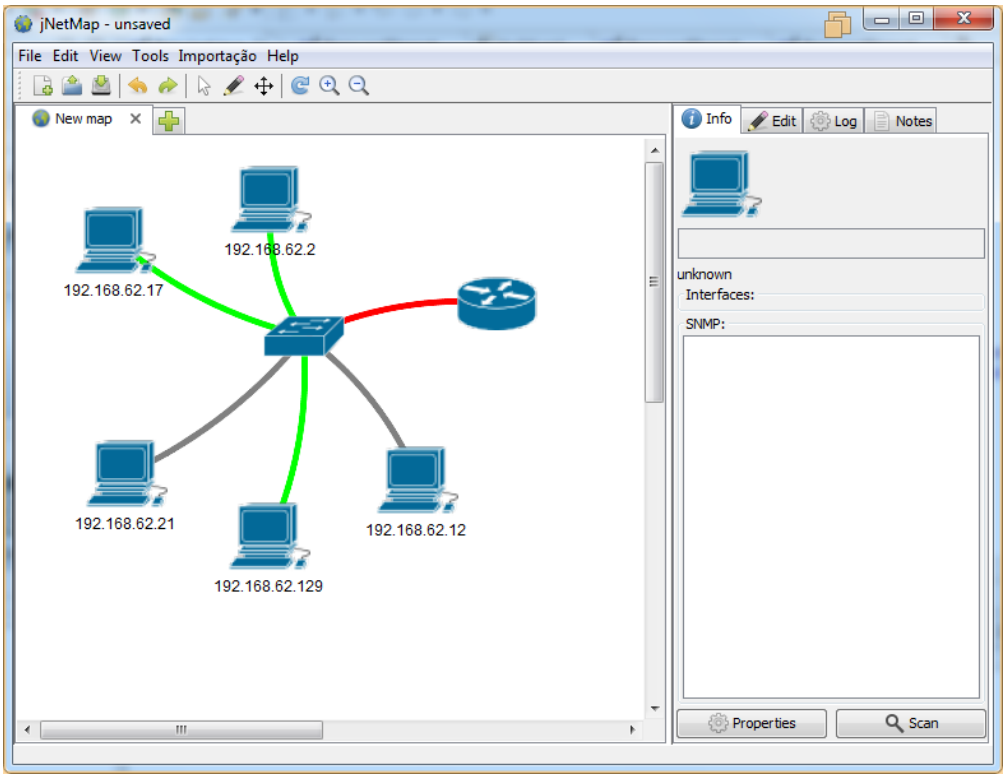


Escaneamento automático de rede (passo 3).

Após o escaneamento ser concluído, o usuário poderá escolher quais dispositivos ele deseja adicionar ao mapa. Basta selecioná-los na lista e clicar no botão “Add”.



Escaneamento concluído.



Dispositivos inseridos no mapa.

Maiores informações estão disponíveis na documentação oficial (que pode ser acessada no menu Ajuda).

Funcionalidades de Integração

Os tópicos descritos a seguir são funcionalidades que foram adicionadas às ferramentas originais.

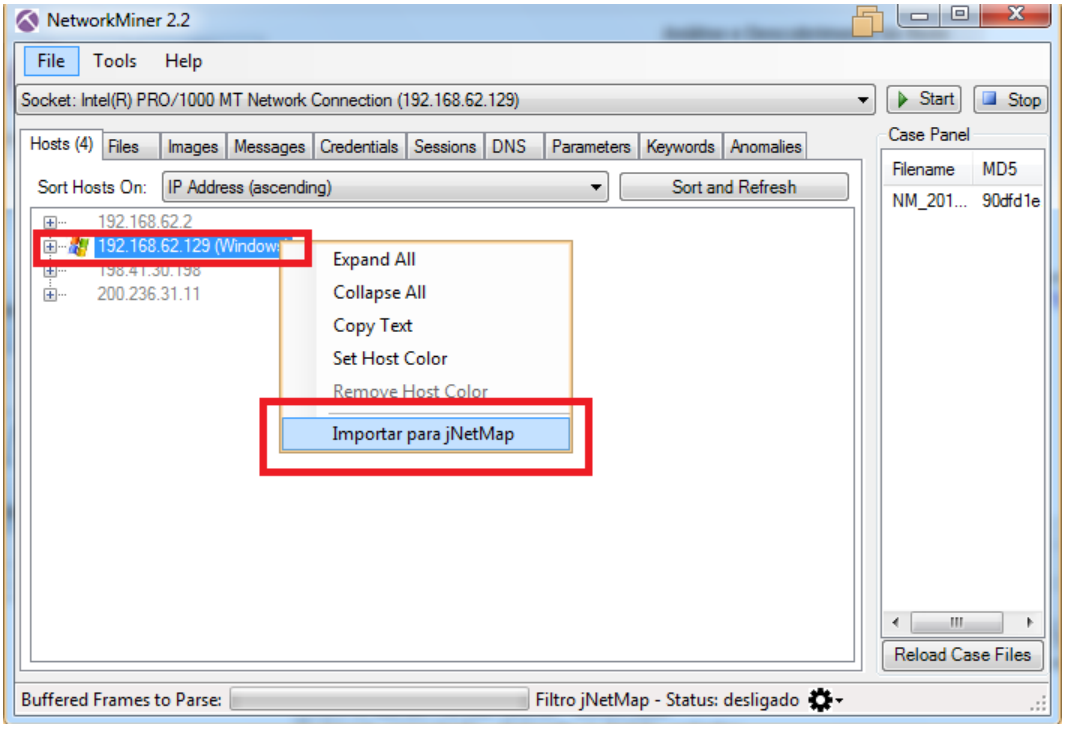
Importar para o jNetMap

O NetworkMiner é capaz de reunir algumas informações sobre os hosts que estão na rede. Entre elas é possível destacar o endereço IP e o sistema operacional do host. A importação de hosts para o jNetMap é uma funcionalidade que envia algumas destas informações do host selecionado para o jNetMap.

Para realizar a importação, basta seguir estes passos:

- Clicar com o botão direto em um host da lista (passo 1)
- Clicar na opção “Importar para jNetMap” (passo 2).
  Neste momento, o jNetMap vai receber as informações enviadas do NetworkMiner.

- Para finalizar a importação é necessário utilizar a ferramenta jNetMap. O usuário deverá escolher em qual mapa deseja importar os dados.



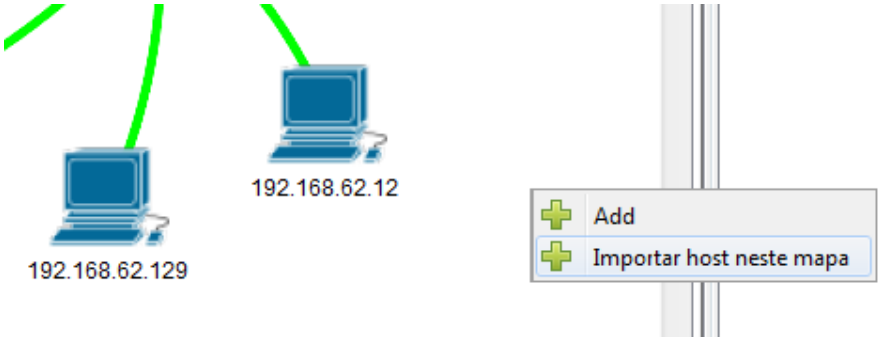
Passo 1 e Passo 2.

Importar host no jNetMap

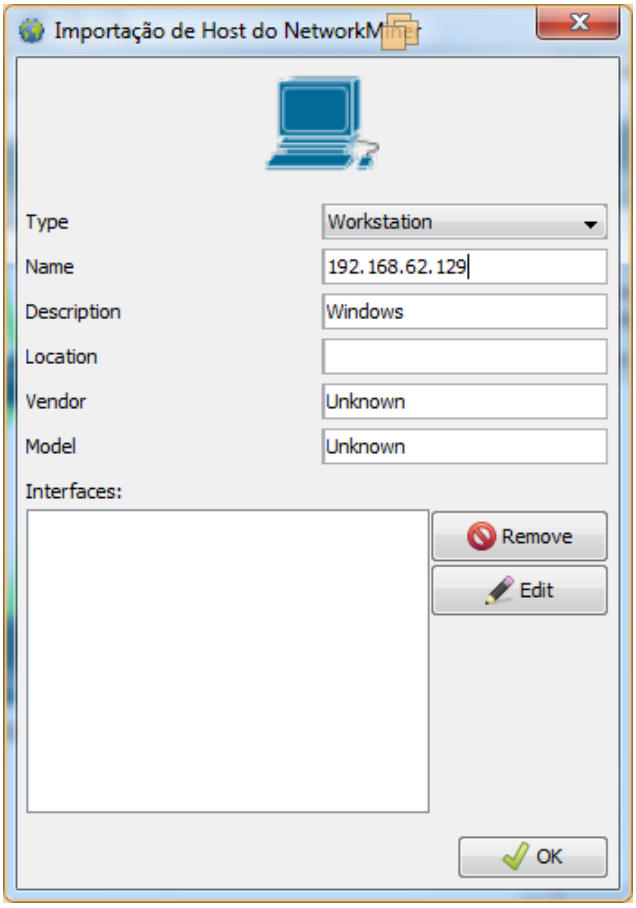
Para importar um dispositivo no jNetMap é preciso previamente enviá-lo do NetworkMiner.

Uma vez que o tutorial “Importar para o jNetMap” (pág 15) tenha sido realizado, os hosts recebidos poderão ser importados para qualquer mapa de rede do jNetMap. O usuário deverá:

- Clicar com o botão direito em qualquer área branca do mapa e escolher a opção “Importar host neste mapa” (passo 1).
- Confirmar e/ou editar as informações recebidas e então confirmar a adição do dispositivo no mapa (passo 2).

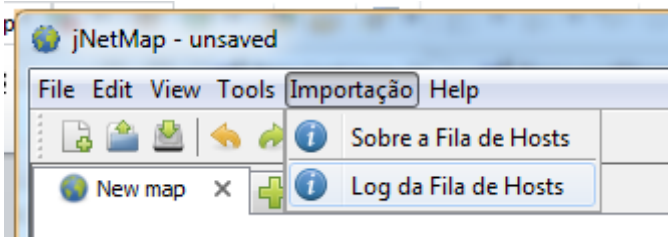


Passo 1.



Passo 2.

O usuário ainda pode utilizar o menu “Importação” para saber mais detalhes sobre este sistema de importação.



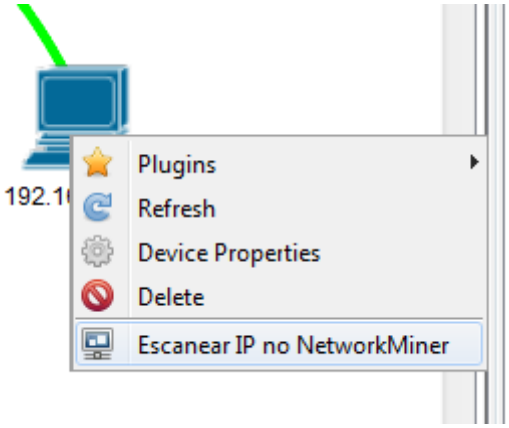
Menu Importação.

Escanear IP no NetworkMiner

O usuário pode enviar um endereço IP para ser escaneado pelo NetworkMiner. Este IP servirá como filtro dos resultados.

Para enviar um endereço IP para o NetworkMiner é necessário:

- Clicar com o botão direito em qualquer dispositivo do mapa (passo 1).
- Clicar na opção “Escanear IP no NetworkMiner”(passo 2).



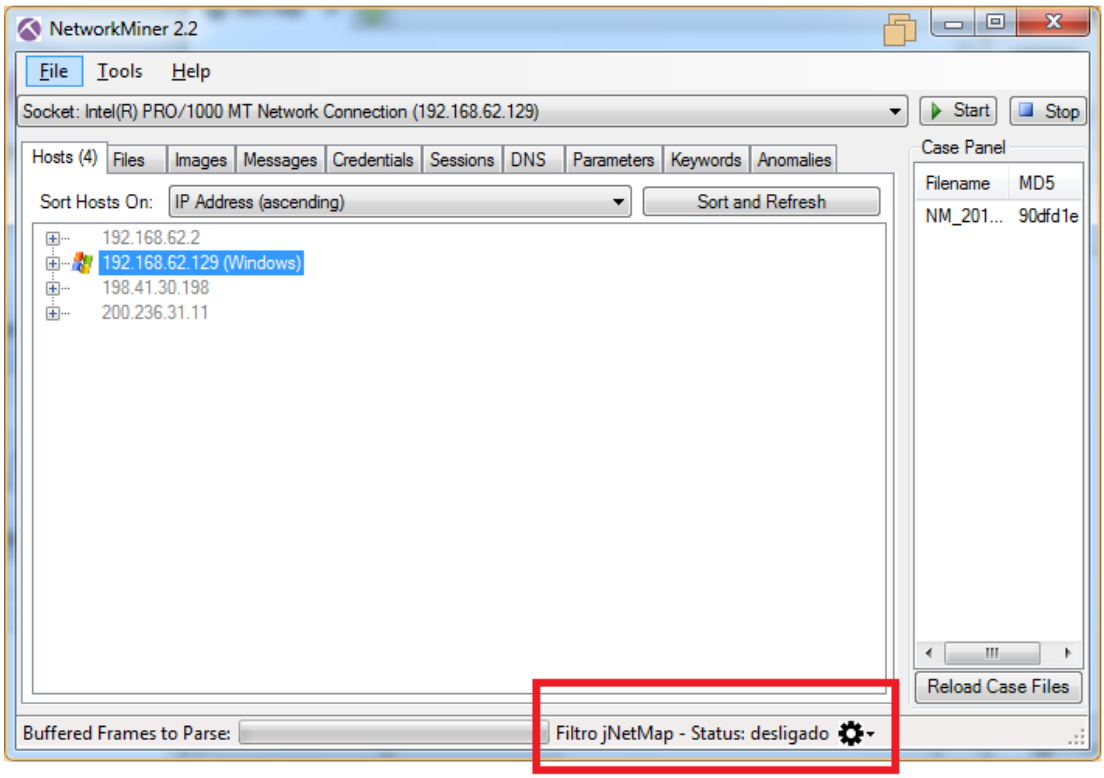
Passo 1 e passo 2.

Automaticamente, o endereço será recebido pelo NetworkMiner e o filtro será ligado.

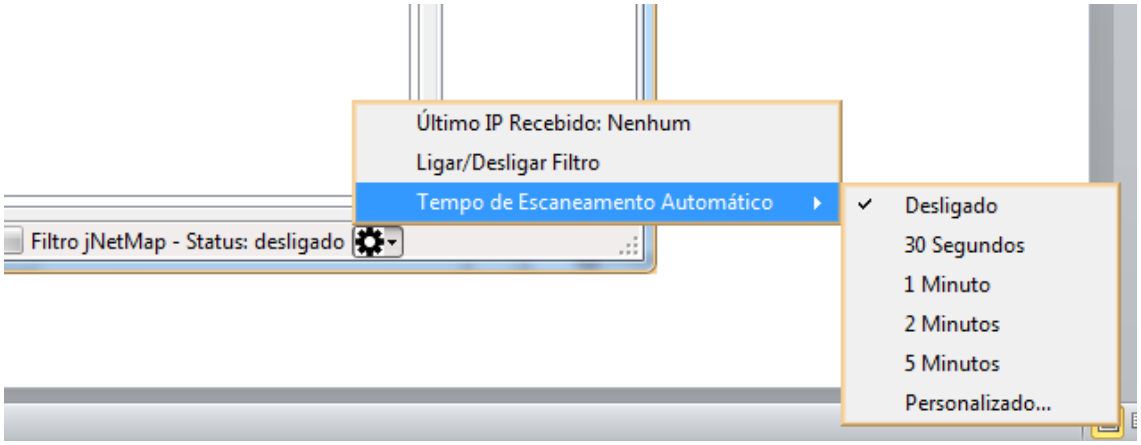
Filtragem de Resultados

A filtragem de resultados é consequência do recebimento de um endereço IP previamente enviado pelo jNetMap. O filtro é automaticamente ativado quando um endereço IP é recebido. Através do menu “Filtro jNetMap” é possível verificar qual IP está sendo utilizado no filtro, ligar/desligar o filtro e definir um intervalo de tempo para o escaneamento automático.

Com o filtro ligado, todos os resultados mostrados na tela serão relacionados com o endereço IP selecionado. Caso o filtro esteja desligado, o escaneamento mostrará todos os resultados obtidos normalmente.



Posição do menu Filtro jNetMap.

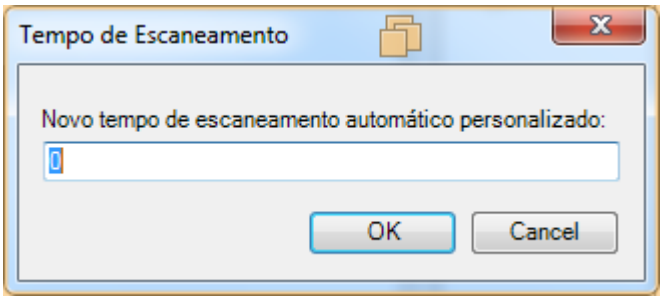


Menu Filtro jNetMap.

Escaneamento **Automático**

O escaneamento automático remove a necessidade do usuário clicar nos botões “Start” e “Stop” para realizar o escaneamento com filtro ao receber um endereço IP do jNetMap.

O escaneamento automático é configurado por padrão como “0 Segundos”. Ou seja, desligado. O usuário pode selecionar entre intervalos pré-definidos ou inserir um intervalo personalizado. Caso queira desativar, basta inserir um intervalo personalizado de 0 (zero) segundos ou clicar na opção “Desligado” dentro do menu “Filtro jNetMap”.



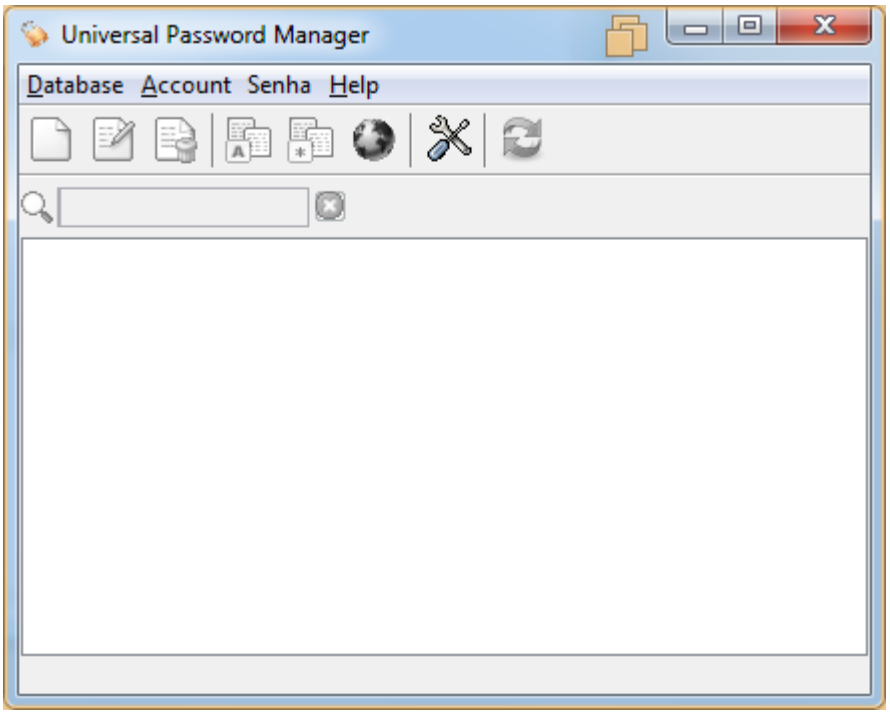
Intervalo personalizado de tempo.

Segurança e Senhas

A categoria Segurança e Senhas engloba duas ferramentas: o Universal Password Manager e o Password Strength Meter. O funcionamento e as funcionalidades de integração dessas ferramentas são detalhados abaixo.

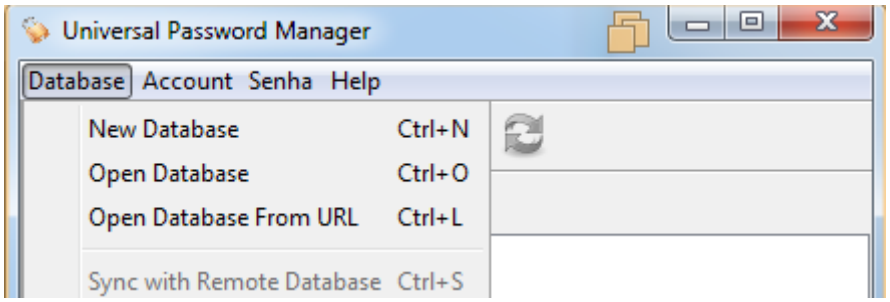
Universal Password Manager

O Universal Password Manager é uma ferramenta para gerenciamento de usuários e senhas, desenvolvida em Java. Ele pode ser acessado pelo menu “Arquivo” ou pelo botão “Universal Password Manager”.



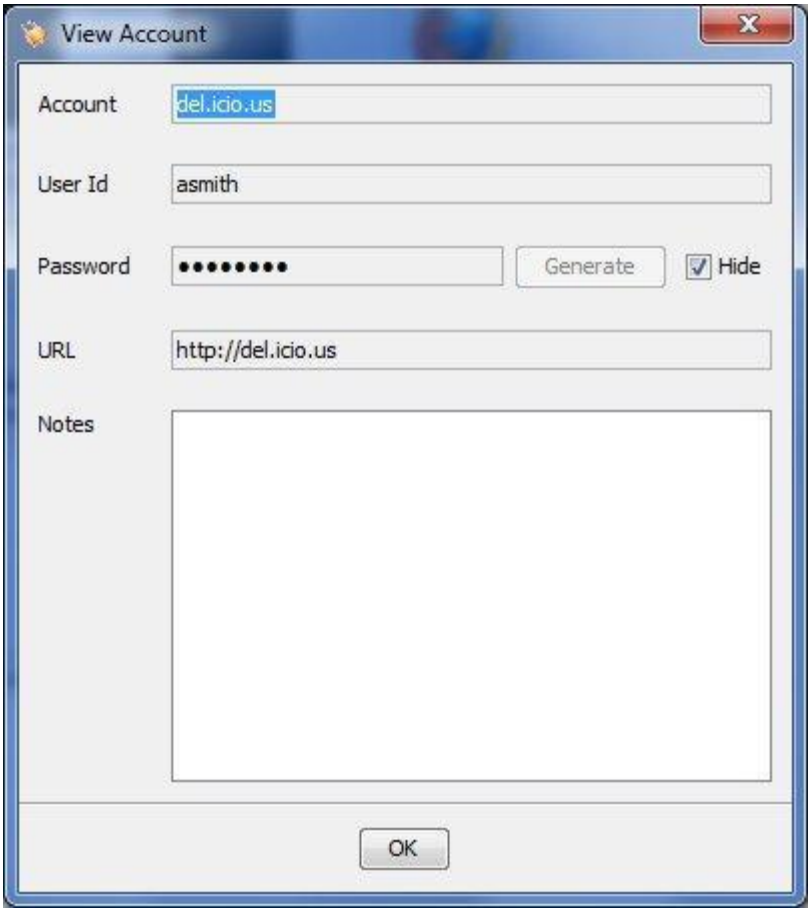
Interface principal do Universal Password Manager Integrado.

Para armazenar e gerenciar as credenciais, é necessário ter uma base de dados. O usuário pode criar uma base nova, ou abrir uma já existente utilizando o menu “Database”.



Menu Database.

No Menu “Account” é possível adicionar novas credenciais de usuário e ver detalhes das já existentes (opção “View Account”). Na interface “View Account” é possível ver detalhes das credenciais, como por exemplo, o usuário, a senha e o endereço de acesso.



Interface “View Account”.

Maiores informações sobre o Universal Password Manager estão disponíveis na documentação oficial, disponível no menu Ajuda da interface principal do protótipo.

Password Strength Meter

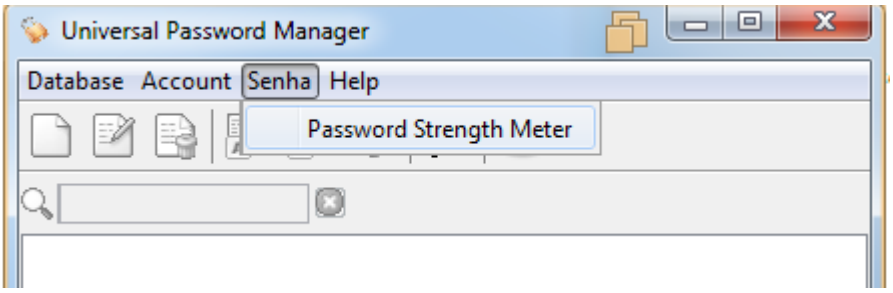
O Password Strength Meter é uma biblioteca desenvolvida em Java. Essa ferramenta é capaz de testar a força das senhas e ainda disponibiliza um número aproximado de iterações necessárias para a quebra da senha. Essa ferramenta pode ser acessada tanto pelo menu “Arquivo”, quanto pelo botão “Password Strength Meter”.

Funcionalidades de Integração

Os tópicos descritos a seguir são funcionalidades que foram adicionadas às ferramentas originais.

Testando uma senha no Universal Password Manager

O usuário do Universal Password Manager poderá testar a força de suas credenciais de rede e/ou outras senhas quaisquer salvas através do menu Senha. Neste menu é possível acessar a ferramenta “Password Strength Meter” e realizar o teste das senhas.



Menu Senha.

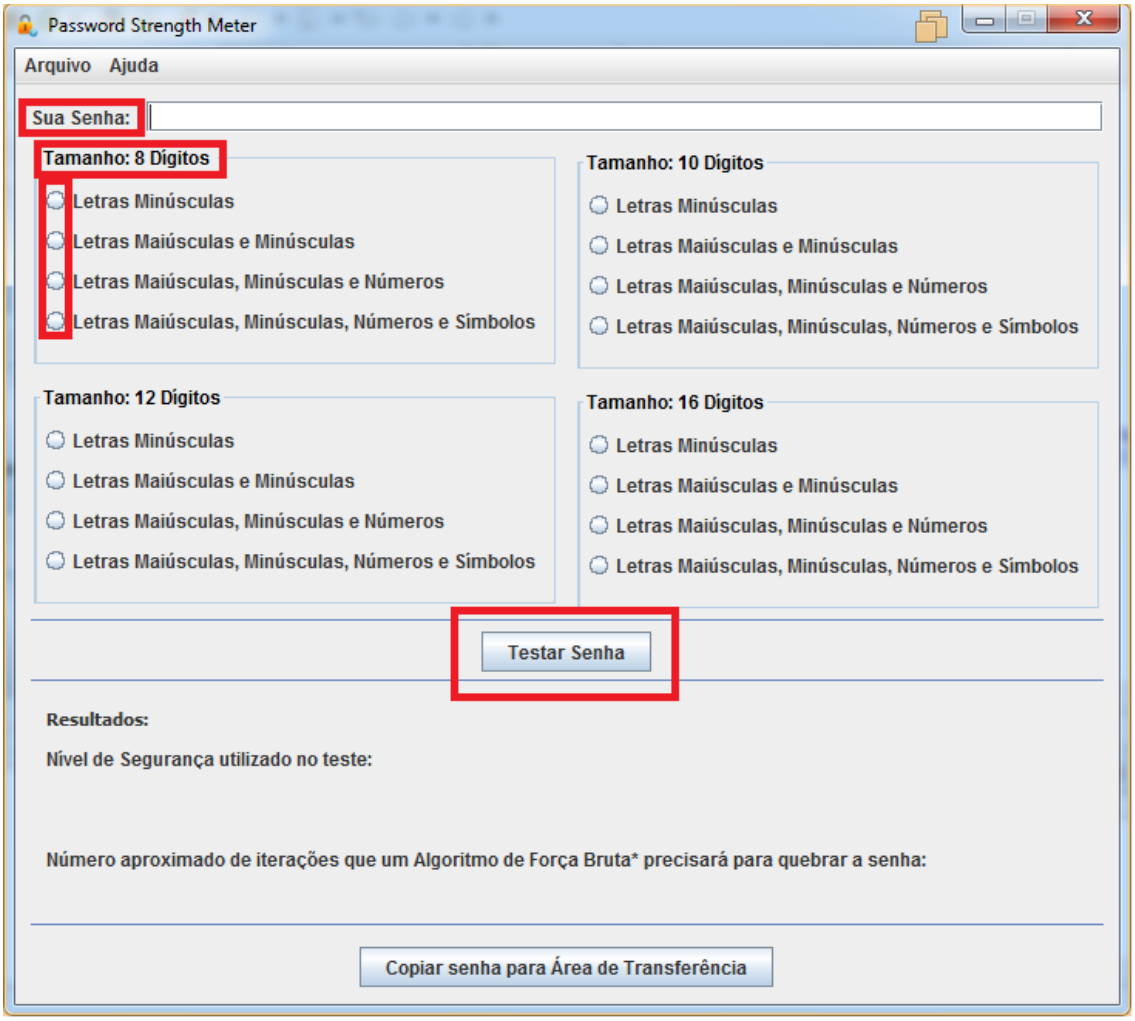
* Observação: Esta pode ser aberta também pelo botão “Password Strength Meter” na interface principal do BYOD Manager Kit.

As informações de como realizar o teste de força de senha são descritas no tutorial “Testando uma senha com o Password Strength Meter”.

Testando uma senha com o Password Strength Meter

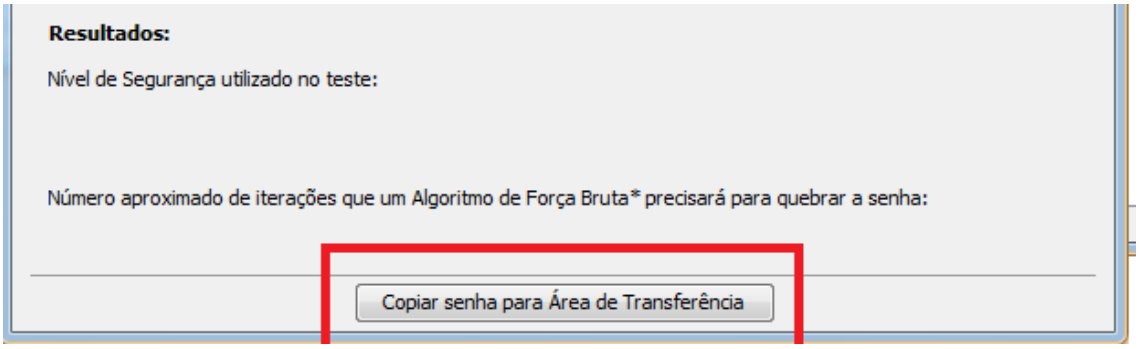
Para testar uma senha é necessário:

- Informar no campo “Sua Senha” a senha que será testada.
- Escolher qual padrão de senha ela deve atender.
- Clicar no botão “Testar Senha”. Os resultados serão exibidos na parte inferior da interface.



Interface principal do Password Strength Meter.

O usuário poderá também copiar a senha para a área de transferência facilmente. Basta clicar no botão “Copiar senha para Área de Transferência”.



Botão “Copiar senha para Área de Transferência”.

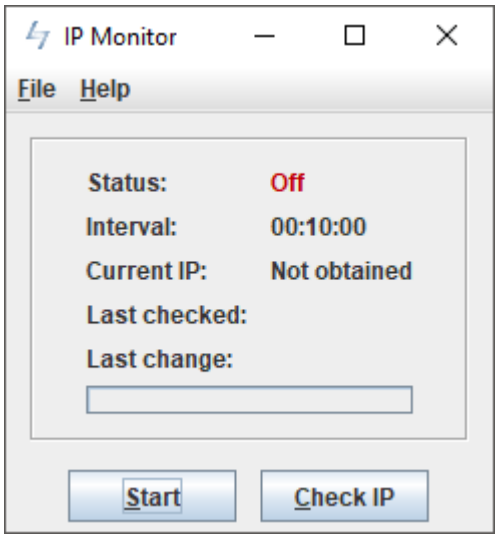
Maiores informações estão disponíveis na documentação oficial do Password Strength Meter.

Administração de Rede

A categoria Administração de Rede engloba duas ferramentas: o IP Monitor e o NetCalculator. O funcionamento e as funcionalidades de integração dessas ferramentas são detalhados abaixo.

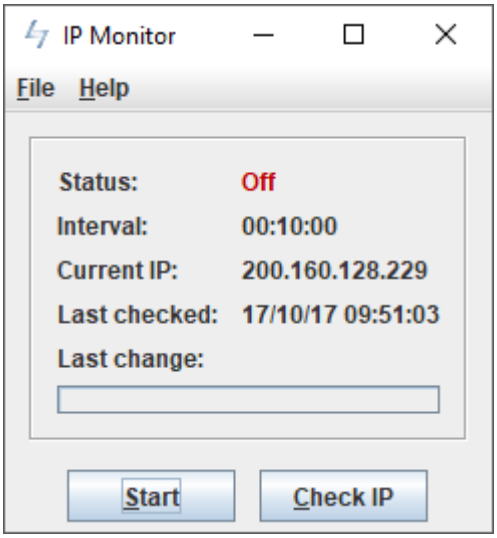
IP Monitor

O IP Monitor é uma ferramenta para monitoramento do endereço de IP público. Ela é desenvolvida em Java e pode ser acessada pelo botão “IP Monitor” ou pelo menu “Arquivo” na interface principal do BYOD Manager Kit.



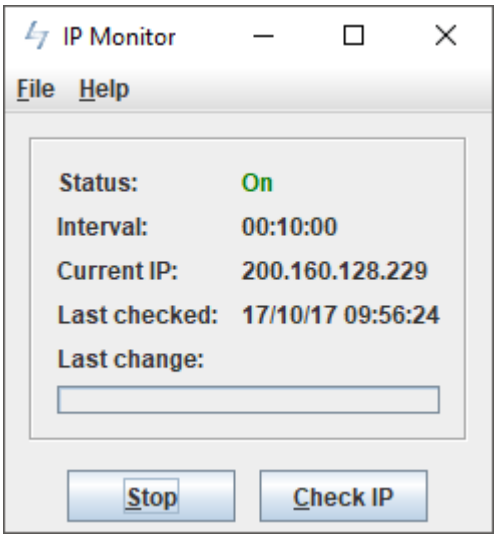
Interface Principal do IP Monitor Integrado.

Para checar o Endereço IP público manualmente, basta clicar no botão “Check IP” na interface principal. O software verificará o endereço e mostrará como resultado o IP atual, o horário da última verificação e o horário da última mudança, caso tenha ocorrido.



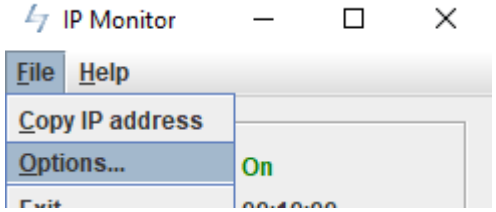
Endereço IP Público verificado às 09:51 do dia 17 de Outubro de 2017.

O usuário poderá iniciar um monitoramento automático do IP público clicando no botão “Start”. Com isso um processo de verificação será inicializado. Este processo consiste em realizar uma verificação periodicamente. Esta verificação informará o IP atual, o horário da última verificação e o horário da última mudança, caso tenha ocorrido.

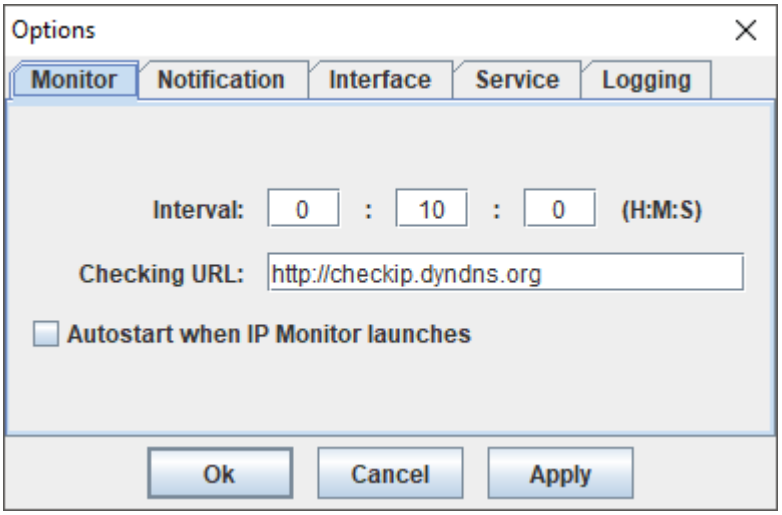


Processo de monitoramento ligado.

O usuário poderá personalizar o intervalo de tempo entre uma verificação e outra, além de definir qual a melhor forma de notificação em caso de mudança de endereço de IP público. Para realizar essa personalização é necessário clicar no menu “File” e então no botão “Options”.



Menu File.



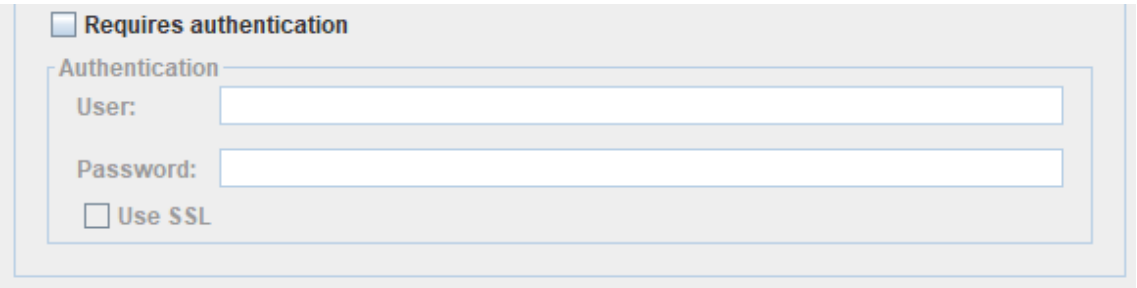
Janela de Opções: Definindo o Intervalo de Escaneamento.

O usuário poderá escolher entre quatro tipos diferentes de notificações. Na aba "Notification" é possível selecionar notificação de áudio, via e-mail, notificação visual ou via linha de comando.

Para efetuar uma notificação por e-mail é necessário:

- Configurar o servidor de e-mail utilizado (passo 1).
- Caso seja necessário, o usuário deve informar usuário e senha (passo 2).
- Completar os campos da Mensagem com Nome, Destinatário, Assunto, etc (passo 3);
- Escrever o corpo da Mensagem (passo 4).

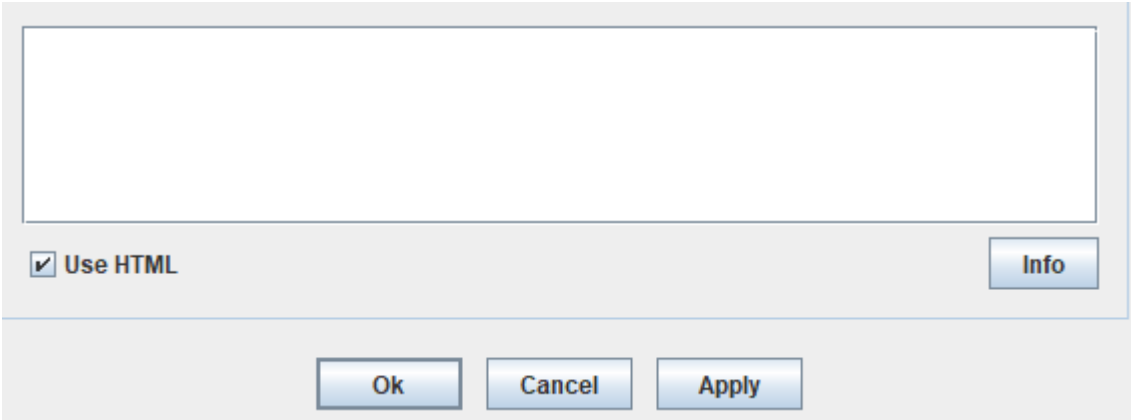
Informar o endereço do servidor de e-mail (passo 1).



Caso necessário, informar o usuário e senha (passo 2).

Dados do e-mail (passo 3).

O Botão “Info” ilustrado na imagem abaixo informa o que o usuário deve escrever no corpo da mensagem para que os dados obtidos pelo IP Monitor sejam corretamente enviados ao destinatário. Estes dados podem ser: IP público antigo, IP Público atual, data e horário do escaneamento, etc...

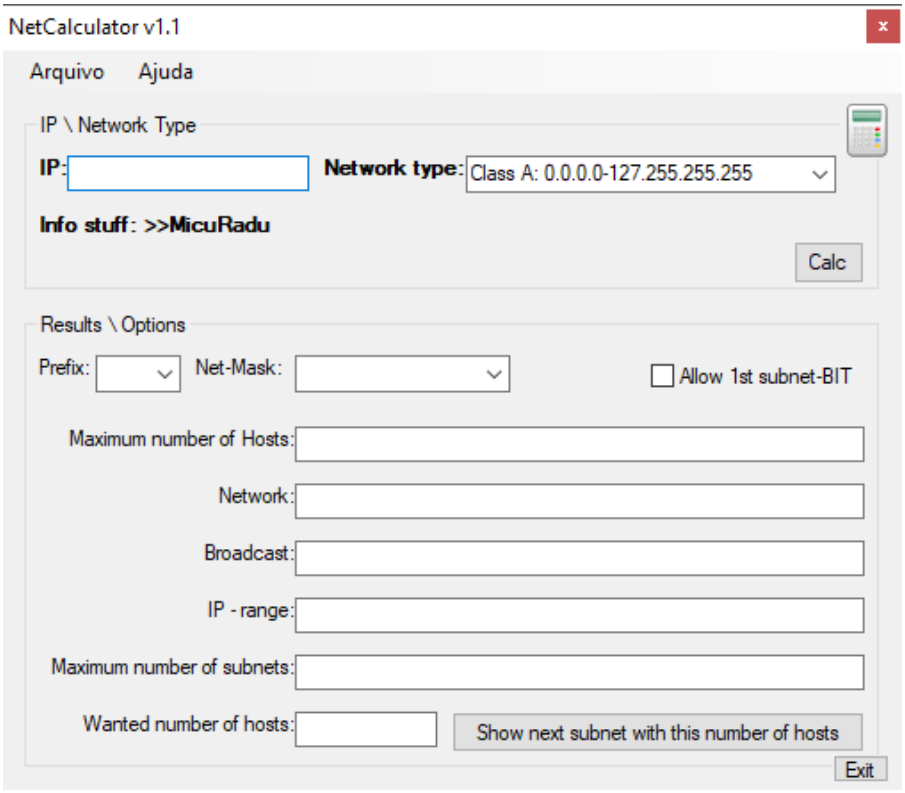


Corpo da Mensagem (passo 4).

Maiores informações sobre a ferramenta IP Monitor estão disponíveis na documentação oficial que pode ser acessada pela interface principal do BYOD Manager Kit.

NetCalculator

O NetCalculator é uma ferramenta desenvolvida em C# para realizar cálculos de rede e subredes. Ela pode ser acessada pelo menu “Arquivo” ou pelo botão “NetCalculator” na interface principal do BYOD Manager Kit.

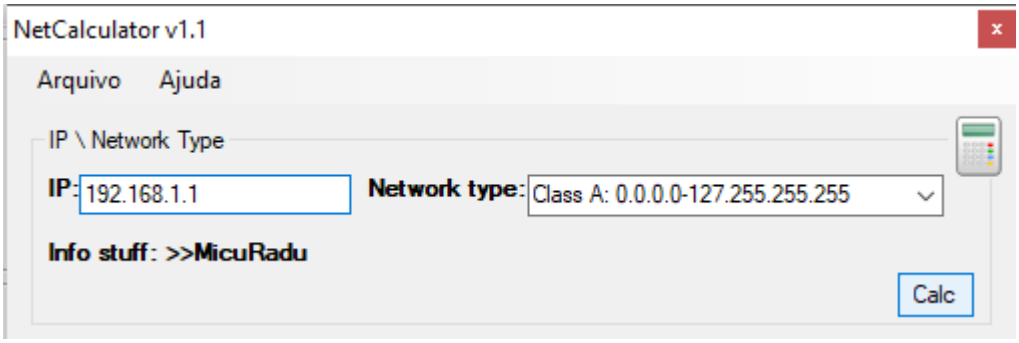


Interface Principal do NetCalculator Integrado.

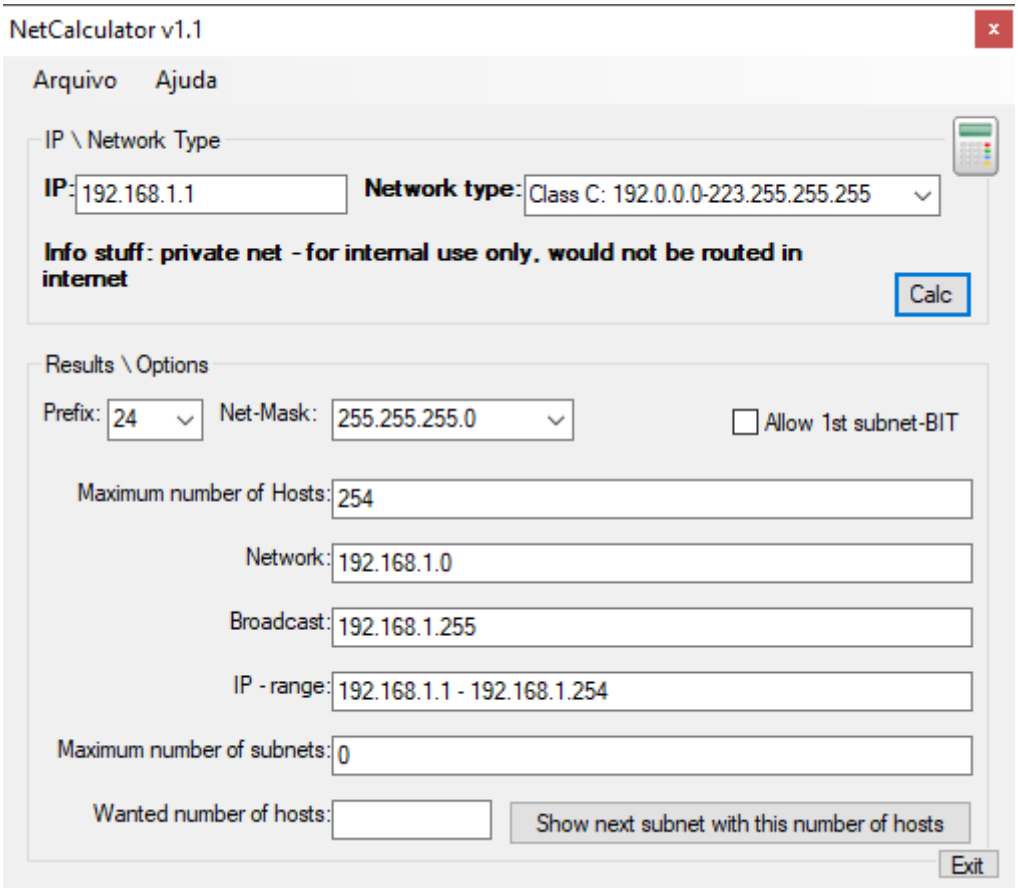
Para realizar o cálculo de rede, é necessário:

- Informar um endereço base no campo IP (passo 1).
- Clicar no botão “Calc” para realizar os cálculos (passo 2).

O NetCalculator vai automaticamente detectar a qual classe o IP pertence e mostrar os resultados na interface.

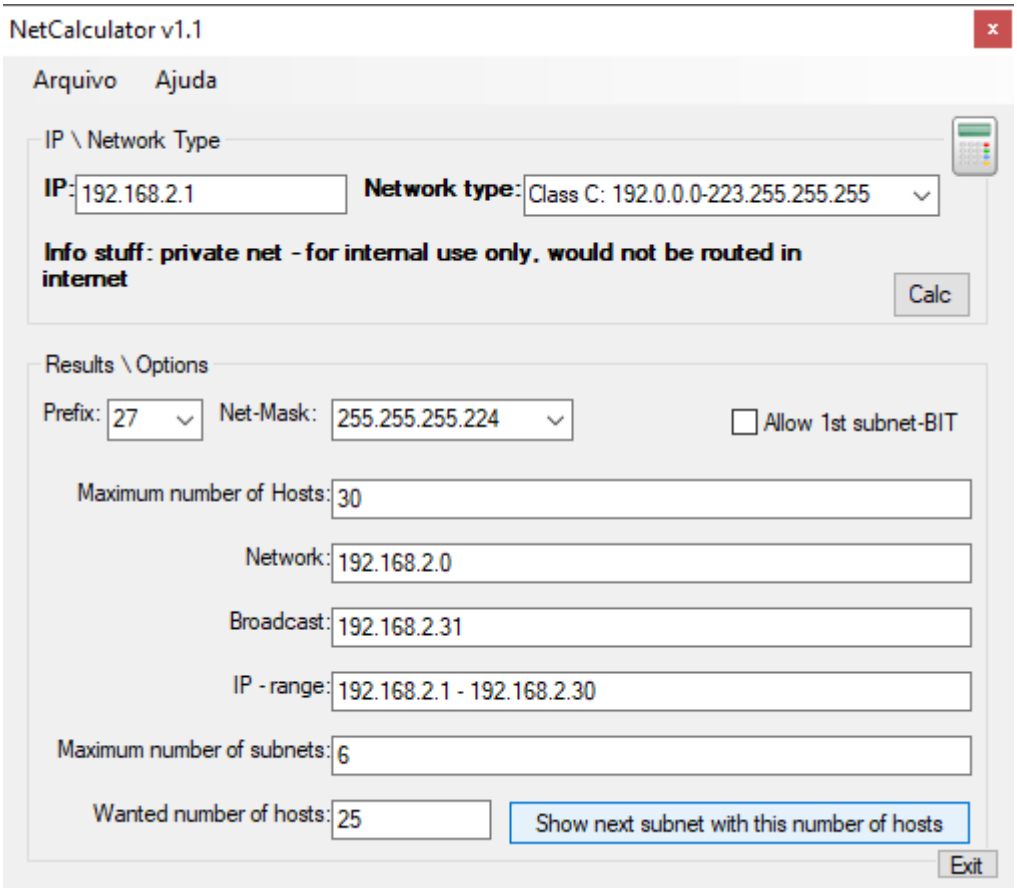


Endereço IP inserido (passo 1).



Cálculo de Rede realizado (passo 2).

Para realizar o cálculo de subredes, o usuário deve informar a quantidade desejada de hosts por subrede e clicar em “Show next subnet with this number of hosts”.



Subrede com 25 hosts.

Mais informações sobre o NetCalculator estão disponíveis na documentação oficial que pode ser acessada pela interface principal do BYOD Manager Kit.

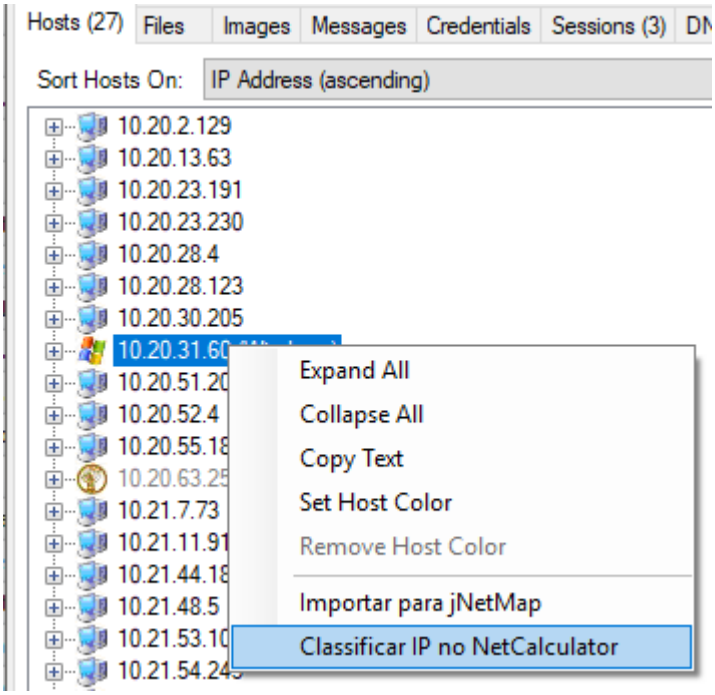
Funcionalidades de Integração

Os tópicos descritos a seguir são funcionalidades que foram adicionadas às ferramentas originais.

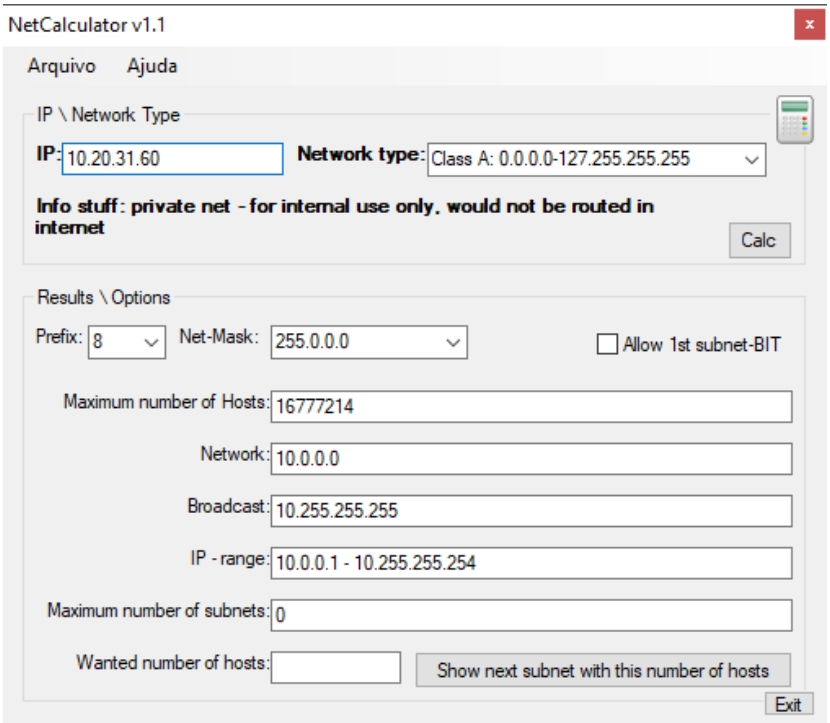
Envio de IP para o NetCalculator

O envio de um endereço IP para o NetCalculator pode ser feito através do NetworkMiner e/ou do jNetMap. Ao receber este endereço IP o NetCalculator irá exibir automaticamente os resultados na interface principal, realizando os cálculos de rede e classificando o IP. Como não é possível descobrir o prefixo de rede utilizando o NetworkMiner ou o jNetMap, o campo “Prefix” precisa ser alterado manualmente.

Para enviar um endereço IP do NetworkMiner para o NetCalculator é necessário clicar com o botão direito do mouse sobre um host previamente listado e então clicar no botão “Classificar IP no NetCalculator”.



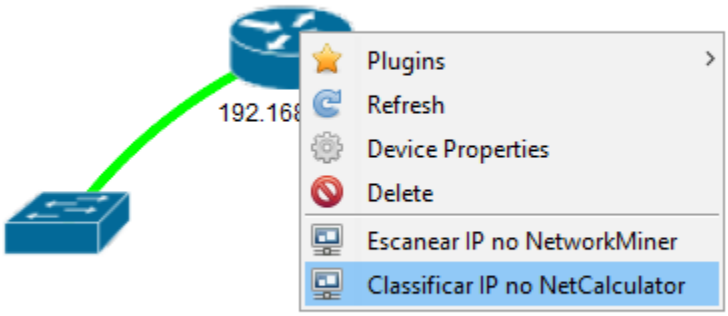
Enviando IP do NetworkMiner para o NetCalculator.



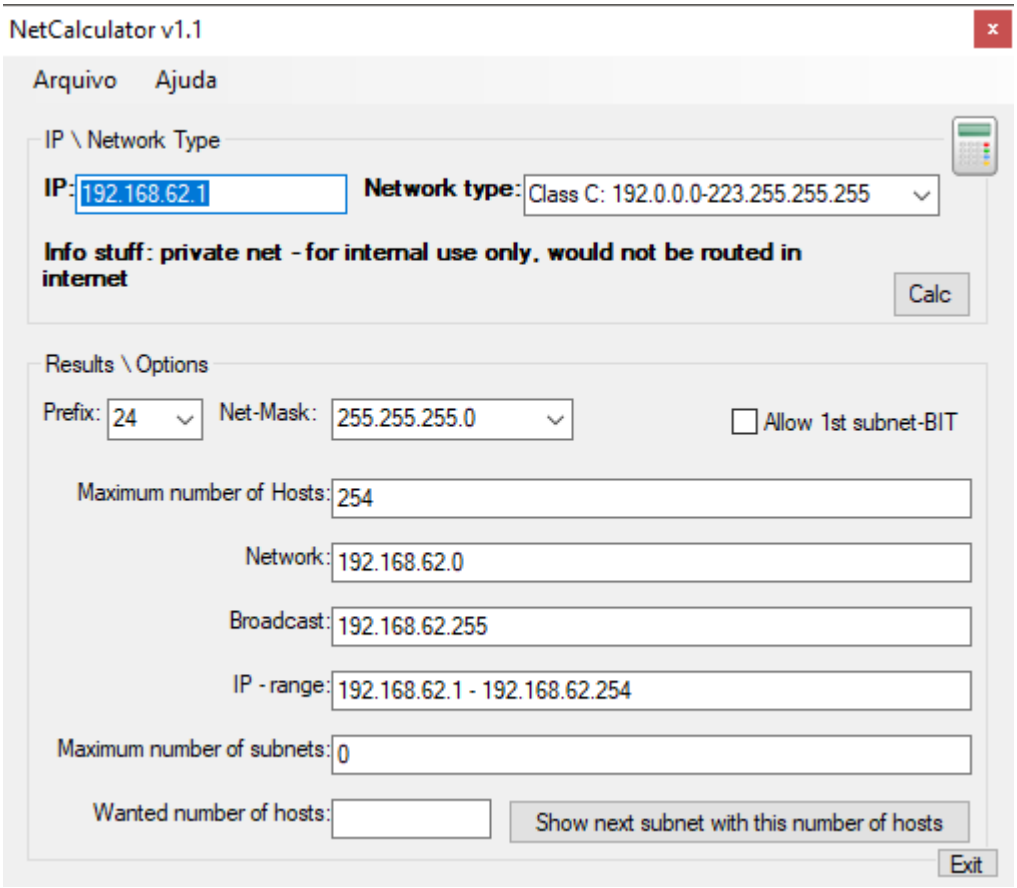
Endereço IP recebido e calculado no NetCalculator.

OBS: O prefixo pode ser alterado manualmente pelo usuário no campo “Prefix”.

Para enviar um endereço IP do jNetMap para o NetCalculator o usuário deve clicar com o botão direito do mouse sobre um dispositivo presente no mapa e então clicar no botão “Classificar IP no NetCalculator”.



Enviando IP do jNetMap para o NetCalculator.



Endereço IP recebido e calculado no NetCalculator.

OBS: O prefixo pode ser alterado manualmente pelo usuário no campo “Prefix”.