

ESPECIAL: Perícia com FTK Imager

Analizando evidências digitais com uma plataforma de baixo custo



Edição 14 :: Ano II



Introdução ao Wireshark

Saiba como avaliar o tráfego da rede

Pentest – Da teoria à prática

Conceitos básicos para
evitar erros na contratação

Segurança em Virtual Private Servers Linux

Uma abordagem analítica e comparativa
de ferramentas open source

STORAGE CORPORATIVO

Entendendo as
siglas e o que
está por trás delas



SEU CARRO TEM SEGURO, SUA SAÚDE TEM SEGURO, MAS E O SEU EMPREGO... TÁ SEGURO??



NÃO DEIXE JUSTAMENTE A SUA CARREIRA FICAR EM RISCO!

Manter-se atualizado com todas as novidades do mercado de desenvolvimento é obrigação de todo bom programador. Faça agora mesmo um seguro para a sua carreira. Seja um assinante MVP!

Saia do risco!



QUEM TEM ESTÁ TRANQUILO.



DEVMEDIA

TENHA ACESSO A:



+DE **260** CURSOS ONLINE



09 REVISTAS MENSAIS



7.850 VÍDEO-AULAS

POR
APENAS **59,90**
MENSAL

Acesse: www.devmedia.com.br/mvp



Ano 2 • Edição 14 • 2013



Assine agora e tenha acesso a todo o conteúdo da DevMedia: www.devmedia.com.br/mvp

EXPEDIENTE

Editor

Eduardo Spinola (eduspinola@gmail.com)

Sub Editores

Marco Antônio Pereira Araújo (maraujo@devmedia.com.br)

Rodrigo Oliveira Spinola (rodrigo.devmedia@gmail.com)

Consultora Técnica

Daniella Costa (daniella@devmedia.com.br)

Jornalista Responsável Kaline Dolabella - JP24185

Capa e Diagramação Romulo Araujo

Atendimento ao leitor

A DevMedia possui uma Central de Atendimento on-line, onde você pode tirar suas dúvidas sobre serviços, enviar críticas e sugestões e falar com um de nossos atendentes. Através da nossa central também é possível alterar dados cadastrais, consultar o status de assinaturas e conferir a data de envio de suas revistas. Acesse www.devmedia.com.br/central, ou se preferir entre em contato conosco através do telefone 21 3382-5038.

Publicidade

publicidade@devmedia.com.br – 21 3382-5038

Anúncios – Anunciando nas publicações e nos sites do Grupo DevMedia, você divulga sua marca ou produto para mais de 100 mil desenvolvedores de todo o Brasil, em mais de 200 cidades. Solicite nossos Media Kits, com detalhes sobre preços e formatos de anúncios.

Fale com o Editor!

É muito importante para a equipe saber o que você está achando da revista: que tipo de artigo você gostaria de ler, que artigo você mais gostou e qual artigo você menos gostou. Fique a vontade para entrar em contato com os editores e dar a sua sugestão!

Se você estiver interessado em publicar um artigo na revista ou no site Infra Magazine, entre em contato com o editor, informando o título e mini-resumo do tema que você gostaria de publicar:



EDUARDO OLIVEIRA SPÍNOLA

eduspinola.wordpress.com

[@eduspinola](https://twitter.com/eduspinola) / [@Java_Magazine](https://twitter.com/Java_Magazine)

Artigo no estilo Solução Completa

04 – FTK Imager: como iniciar uma Perícia

[*Marcelo Lau e Nichols Jasper*]

11 – Desmistificando o Storage Corporativo

[*Mateus Espadoto*]

19 – Analisando o tráfego de redes com Wireshark

[*Christiane Borges Santos*]

Conteúdo sobre Boas Práticas

26 – Pentest: avaliação do nível de segurança de uma rede

[*Roberto Henrique*]

Conteúdo sobre Boas Práticas

33 – Segurança de servidores Linux com ferramentas open source

[*Thiago José Lucas e André Domingues*]

Sumário

FTK Imager: como iniciar uma Perícia

Saiba como analisar uma evidência digital com uma plataforma de baixo custo

Hoje em dia os dispositivos móveis são usados em todos os lugares, tanto que os profissionais de TI costumam possuir pelo menos um dispositivo de armazenamento USB para realizar o backup de seus arquivos ou para usá-los em outro computador, como uma forma prática para transferência de arquivos.

Nesse cenário, sabemos que este tipo de mídia é suscetível ao extravio devido ao tamanho físico reduzido, resultando na perda dos dados e provavelmente muitas dores de cabeça. Com a popularização destas mídias de armazenamento e a computação ubíqua, um perito forense precisa ser capaz de analisar artefatos forenses como esses, que podem ser essenciais para o contexto de uma investigação.

Sabendo da popularização destas mídias e de seu baixo custo de aquisição, elas se tornam uma das principais preocupações contra o vazamento de dados nas empresas, ameaça que procura ser mitigada com controles tecnológicos, como soluções de *Endpoint Protection* e *Data Leak Protection (DLP)*. Como nem todas as empresas conseguem aplicar e gerenciar eficientemente estes controles, é essencial que um perito forense conheça as técnicas para preservar e analisar este tipo evidência, pois elas serão cada vez mais frequentes nas investigações computacionais, principalmente com usuários que tentam evadir informações das empresas por meio de mídias removíveis.

Este usuário mal intencionado pode tentar apagar arquivos desta mídia para destruir possíveis evidências de seus atos. Porém, quando um usuário realiza uma formatação lógica, ele nunca terá certeza se os seus dados ou segredos serão detectáveis por outras pessoas. Demonstraremos que, no sistema de arquivos NTFS, essa certeza da não recuperação dos dados não ocorre sem o uso de um método de limpeza que sobrescreva adequadamente as áreas físicas do disco onde os dados são

Resumo DevMan

Porque este artigo é útil:

Este artigo aborda um estudo de caso com o uso da ferramenta FTK Imager. Neste estudo de caso, considera-se um cenário hipotético, onde um pendrive é encontrado com um suspeito e parece estar sem informações quando acessado pelo computador. Diante deste cenário, apresentaremos o processo de preservação de uma evidência digital com validade investigativa aplicada ao sistema de arquivos de um pendrive por meio da ferramenta FTK Imager, que também nos auxiliará a encontrar artefatos excluídos no dispositivo.

Os pontos destacados neste artigo serão úteis para melhorar a compreensão de um perito em forense computacional quanto à importância de se conhecer os sistemas de arquivos e como utilizar uma ferramenta para preservar uma evidência digital e analisá-la posteriormente.

armazenados. Além disso, podemos recuperar valiosas informações de um disco de armazenamento com o intuito de comprovar uma dada hipótese e resolver uma determinada investigação com o uso de ferramentas apropriadas.

A seguir descreveremos o cenário pericial abordado neste artigo.

Contextualização do cenário

Um perito forense irá se deparar com os mais variados cenários investigativos, com contextos, variáveis e dificuldades particulares a cada situação. No exemplo deste artigo, o perito é requisitado para ajudar em uma investigação envolvendo a cópia de arquivos confidenciais.

Após uma básica introdução do caso feita pelos gerentes, o perito se dirige ao local de trabalho de um suspeito de evadir informações, comprometendo a propriedade intelectual da empresa. Ao entrar no local de trabalho do suspeito, o perito o visualiza digital rapidamente alguns comandos em uma estação Windows 8. A partir dessa atitude suspeita, o perito solicita que o suspeito se afaste da estação de trabalho para que ele possa examiná-la

em busca de possíveis evidências de atividade ilícita. Quando o perito começa a examinar o computador, visualiza apenas um pendrive sem conteúdo aparente no Windows Explorer, conforme a **Figura 1**.



Figura 1. Conteúdo visualizado no Windows Explorer

Considerando o cenário e as suspeitas existentes, devemos formular hipóteses que nos ajudem na condução do caso, as chamadas linhas de investigação. Se nosso suspeito eliminou os arquivos, de que forma isto pode ter acontecido? A partir de técnicas de Wipe? Ocultação de arquivos? Alguma ferramenta de armazenamento em nuvem? Uso do comando [Ctrl+x] para recortar os arquivos?

As linhas devem ser analisadas e investigadas conforme a sua probabilidade e relevância para o caso.

Metodologia e estratégia investigativa

O perito pode até ter uma boa ideia do que aconteceu, mas como provar isso perante a justiça? Para provar os fatos de acordo com uma linha investigativa é necessário se fundamentar em outras evidências, já que o ambiente de TI desta empresa não possui certos controles aplicados para apoiar o argumento de fraude corporativa, como a existência de logs de sistema ou uma monitoração contínua das atividades dos usuários.

É preciso apresentar ao juiz um relatório pericial com argumentos embasados em provas, com uma boa organização e que possibilite a reconstituição dos fatos com o mínimo grau de incerteza, pois qualquer dúvida sobre o que aconteceu será um ponto onde o advogado de defesa tentará fragilizar seu relatório e suas evidências, fortalecendo o réu perante a corte.

Para fins didáticos, vamos apresentar uma maneira de condução pericial por meio da busca de arquivos deletados, seguindo a linha de raciocínio de que nosso suspeito deletou os arquivos utilizando o comando [Ctrl+x] ou [Shift+Del] para limpar o seu pendrive.

Não devemos nos esquecer das situações em que a preservação da evidência necessita de fé pública, isto é, quando é necessária sua apresentação perante um foro judicial, sendo requerida a presença de um escrivão para acompanhamento do processo mencionado, visando à produção de uma ata notarial que descreva os fatos ocorridos.

Para facilitar a descrição dos fatos e possibilitar uma avaliação da conduta do perito – quando nomeado pelo juiz – ou assistente técnico, caso este profissional seja contratado pela parte (requerido ou requerente), é necessário que seja seguida uma metodologia de trabalho reconhecidamente aceita e difundida na área pericial como

boa prática de trabalho. Uma delas é a RFC 3227 (veja a **BOX 1**), intitulada *Guidelines for Evidence Collection and Archiving*.

A importância da RFC 3227 é a orientação da prática investigativa, descrevendo para o perito que ele deve seguir alguns princípios e uma ordem de ações que precisam ser respeitadas.

BOX 1. RFC

RFC – Request for Comments – é um conjunto de documentos de referência junto à comunidade da Internet que descreve, especifica, padroniza e debate a maioria das normas, padrões, tecnologias e protocolos ligados às redes e à computação.

De acordo com a RFC 3227, uma evidência deve ser identificada, preservada, analisada e apresentada. Estes quatro princípios forenses consideram que não se pode apresentar o que não foi analisado, não se pode analisar o que não foi devidamente preservado e não se pode preservar o que não foi identificado. Qualquer quebra de confiança nestas atividades relacionadas pode colocar a perder qualquer iniciativa pericial, mesmo que o profissional esteja habilitado a esta atividade e utilize as ferramentas e procedimentos adequados, pois por mais que um perito ou assistente técnico esteja acostumado com suas atividades, o menor descuido de sua parte pode fazer com que se torne responsável pela perda de confiabilidade de um trabalho composto por muitas dezenas de horas.

Não é por menos que estes processos investigativos são mais bem tratados quando o perito ou assistente técnico é devidamente treinado nesta “arte”. Hoje há diversas formações, sejam estas de curta duração, fornecidas presencialmente ou à distância, com uma variada quantidade de horas, até cursos de pós-graduação stricto ou lato sensu. Entretanto deve-se deixar claro que uma formação acadêmica, ou mesmo uma certificação, é apenas um passo para aquisição de conhecimento. Para que um profissional realmente se torne um especialista investigativo em crimes eletrônicos é necessária à prática constante e a disciplina em aprender e desenvolver novos conhecimentos, visto que essa área sofre constante mudança devido às tendências e inserção de novas tecnologias.

Para os leitores do artigo, este é o momento de identificar sua habilidade em atuar nesta área ou não. O domínio de uma ferramenta como o FTK Imager é apenas uma destas habilidades, mas a principal característica desejada a um perito é sua capacidade em relatar o que foi identificado por ferramentas técnicas em linguagem não técnica, ou seja, uma linguagem coloquial, evitando o uso de termos somente técnicos. E mesmo quando os termos técnicos são necessários, considera-se importante que o profissional em perícia seja capaz de relatar todo o processo de forma clara e concisa, considerando que o receptor deste laudo possivelmente será alguém leigo em tecnologia e que não conhece todos os detalhes do conteúdo existente na evidência analisada.

Algo que os profissionais ainda devem estar cientes, é que dificilmente uma atividade investigativa será conduzida apenas por uma única ferramenta, sendo necessário o conhecimento de uma

série de ferramentas aplicáveis a casos e contextos particulares. Mencionamos esta questão até para deixar claro que se o processo investigativo fosse simples, ele seria facilmente automatizado e se tornaria autossuficiente, não necessitando mais da interferência humana para efetivação da perícia.

O conhecimento de diversas ferramentas pode ajudar um perito a despendar menos tempo na análise das evidências devido à grande quantidade de dados que precisam ser processados. Portanto, se não houver um processo de triagem e priorização dos artefatos, possivelmente o perito ou assistente técnico perderá um tempo precioso que poderia ser mais bem utilizado durante uma investigação.

Outro ponto é o conhecimento de ferramentas periciais para diversas plataformas. Apesar do caso aqui exemplificado se tratar de uma coleta de evidência em ambiente pericial Windows, devemos estar cientes de que as ferramentas da área investigativa estão disponíveis tanto para ambientes Windows, quanto Linux/Unix, onde estão disponíveis tanto versões gratuitas (*open source*) quanto comerciais para ambas as plataformas.

As evidências também não estão restritas a computadores pessoais, servidores de trabalho e mídias removíveis. Hoje temos como evidências os elementos de rede, tais como roteadores, firewalls, impressoras, dispositivos móveis como telefones celulares, smartphones, tablets, entre outros, vindo até a abranger dispositivos de uso doméstico como Smart TVs e consoles de jogos (Wii, PlayStation, entre outros), que podem agregar dados importantes em um contexto investigativo.

Com este breve descritivo metodológico, ressaltamos a importância de o perito ou assistente técnico se atualizar constantemente sobre novas técnicas e tecnologias. A partir de agora, vamos proceder com a continuidade deste estudo de caso, mostrando como um profissional pode iniciar sua primeira perícia.

Visão técnica e resposta ao incidente

Analisar o ocorrido neste estudo de caso a partir de uma perspectiva técnica irá acrescentar informações úteis para validar a linha investigativa escolhida. O suspeito estava usando um sistema Windows 8 como estação de trabalho e foi informado que ele estava coletando ilegalmente informações da empresa, mas não se sabe como isso ocorreu. Quando o perito acompanhado do supervisor entrou no local para conversar com o suspeito, ele rapidamente inseriu comandos em sua estação de trabalho. Tais comandos irão mostrar se as ações por ele executadas eram legais ou se estava tentando esconder algo.

Primeiramente o suspeito é levado para longe de sua estação de trabalho, para junto de outros colaboradores responsáveis. Ao examinar a estação de trabalho, o perito percebe que ela não estava conectada a nenhuma rede, nenhum cabo Ethernet ou dispositivo sem fio, voltando suas atenções para o sistema operacional e particularidades do sistema de arquivos, neste caso, o NTFS.

Depois de concluído que o tráfego de rede não é o melhor caminho para seguir com esta linha investigativa, é feita uma imagem do sistema ainda vivo, isto é, em funcionamento, para

salvar os processos em execução e outras informações voláteis utilizando-se de um sistema de análise forense, como a distribuição CAINE, e em seguida é desligada a estação de trabalho. Na sequência, é utilizado um duplicador de disco (como o *Forensic Duplicator* da Tableau) para garantir que uma cópia do disco rígido e do pendrive seja feita com verificação de integridade e admissibilidade jurídica.

No laboratório forense, é feita outra cópia da primeira duplicação para início da análise, além de uma segunda cópia, pois se recomenda que o profissional investigativo sempre tenha consigo mais de uma cópia da evidência original. Assim, em caso de perda de integridade de uma das cópias, ainda será possível prosseguir com a atividade de análise de evidência por meio da outra cópia preservada. Para manter a integridade entre as cópias, deve-se certificar se o valor do hash da cópia da primeira duplicação é mantido para garantir sua admissibilidade futura durante um processo judicial. Os algoritmos de hash (veja o **BOX 2**) geralmente utilizados e reconhecidos por sua segurança são os da família SHA (*Secure Hash Algorithm*), em especial o SHA1, cuja probabilidade de geração de mesma combinação por outra evidência equivalente (evento chamado de colisão) é quase nula.

BOX 2. Função Hash

Uma função hash é um algoritmo que gera uma sequência de bits a partir de uma entrada de dados. Esta sequência de saída busca identificar um arquivo ou informação unicamente.

Seguindo a linha investigativa adotada, se o suspeito vier a apagar arquivos, o sistema de arquivos NTFS executa o seguinte processo: marca os arquivos como excluídos dentro da entrada da MFT (*Master File Table*), deixando a sua posição na MFT pronta para ser reutilizada pelo sistema operacional. Portanto, o conteúdo – os bits propriamente ditos – estará lá até que esta posição na tabela MFT do dispositivo de armazenamento seja reescrita e o último conteúdo, sobreposto. Um vídeo do blog *WhereIsYourData* mostra este processo de deleção apenas lógica dos dados, evidenciando que fisicamente os bits ainda estão gravados no disco. O endereço para este vídeo está disponível na seção **Links**.

Prosseguindo com a linha de raciocínio de que houve a deleção de arquivos, o perito prossegue a análise com foco na possibilidade de uso do comando [Ctrl+x]. Este comando é uma combinação de duas operações do sistema NTFS Windows. Em primeiro lugar, há uma cópia dos arquivos para a memória, e quando o local de inserção é selecionado e os arquivos são movidos com êxito, há o segundo passo: a exclusão da referência na tabela de MFT. Em outras palavras, uma ação de exclusão no sistema de arquivos.

Ressaltamos que apesar de o sistema operacional Windows utilizar o NTFS na formatação deste caso, a maioria das mídias removíveis, como pendrives, utilizam a formatação FAT32. Independentemente disso, o processo indicado neste artigo permite a preservação de evidências em ambas as formatações, FAT32 e NTFS.

Para melhor compreender a busca por arquivos excluídos, iremos detalhar a prática de montarmos uma imagem de disco com o intuito de investigar estes arquivos. Para isso, utilizaremos a ferramenta forense FTK Imager para realizar a clonagem do pendrive e visualizar os efeitos da ação de deleção nos sistemas de arquivos predominantes na plataforma Windows.

Duplicação e análise de um pendrive com FTK Imager

O FTK Imager é um software forense criado pela empresa Access Data que possui funcionalidades para criar imagens de disco, realizar despejos (dumps) de memória e até mesmo recursos básicos de análise pericial em imagens de disco nos formatos aceitos pela ferramenta.

Este artigo se valerá da ferramenta FTK Imager por duas razões: ela é gratuita e intuitiva para os entusiastas da área que conhecem o básico sobre computação forense e desejam evoluir seus conhecimentos com investimentos de baixo custo para iniciar seu laboratório pericial.

Dito isso, em primeiro lugar, é necessário fazer o download da ferramenta no site da empresa (veja a seção **Links**). Utilizaremos a versão 3.1.2 do FTK Imager. Após um breve cadastro e com o download realizado, mostraremos o passo a passo de instalação. Ao iniciarmos o executável, veremos a tela apresentada na **Figura 2**.

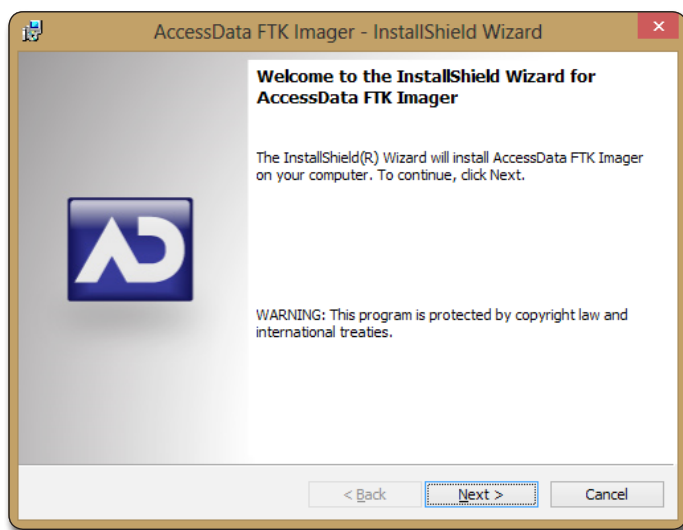


Figura 2. Início da instalação do FTK Imager

O processo de instalação é simples e não requer quase nenhuma configuração adicional; um processo NNF (*Next, Next, Finish*) instalará a ferramenta FTK Imager na estação pericial. Na tela final, não marcaremos a caixa de execução para que possamos preparar a estação forense, visando não alterar a evidência original e comprometer sua admissibilidade durante uma gravação indevida feita na mídia removível. Isto pode acontecer de forma não intencional, pelo uso de algum software instalado no computador ou do próprio sistema operacional.

Para preservar a aceitação da evidência perante um júri e tornar o processo de preservação mais robusto contra possíveis questionamentos, é recomendado o uso de uma solução de hardware ou software para bloquear a permissão de “gravação” nas portas USB do seu computador. A solução de software altera a chave do registro abaixo para garantir que o sistema operacional não “escreva” no dispositivo:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\StorageDevicePolicies\WriteProtect
```

De preferência, deve-se utilizar outro pendrive para testar se a proteção contra gravação foi mesmo habilitada, pois caso exista algum erro na configuração desta chave de registro, a evidência não será comprometida.

Considerando que a porta USB já está protegida, colocamos o pendrive do suspeito na entrada USB e iniciamos a produção da imagem do sistema de arquivos original visando a geração da cópia que será posteriormente analisada. Com o FTK Imager executando com privilégios administrativos (para conseguir manipular todos os dispositivos conectados), a interface da **Figura 3** é exibida.

Agora, no menu *File*, acesse a opção *Create Disk Image*, conforme indica a **Figura 4**.

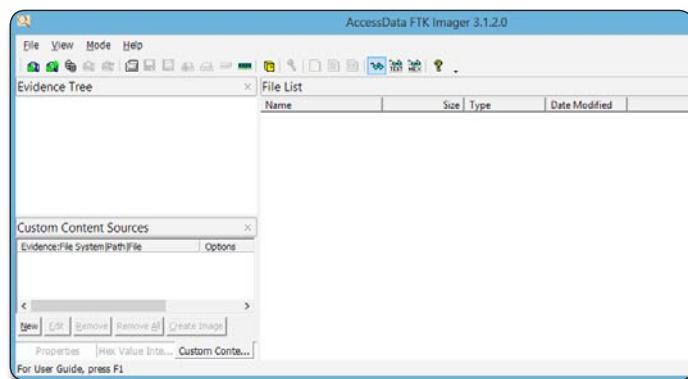


Figura 3. Interface do FTK Imager

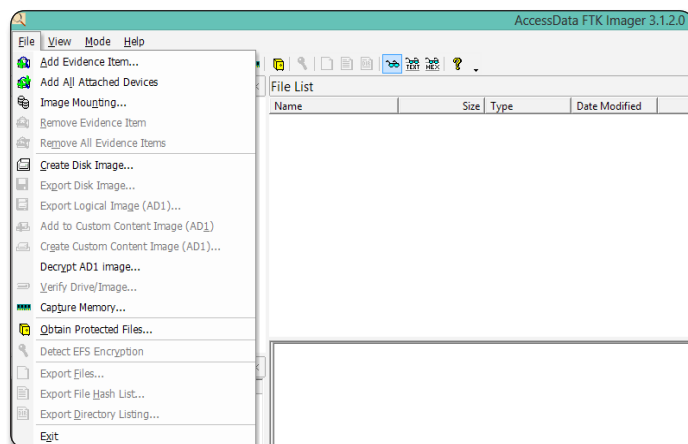


Figura 4. Função Create Disk do FTK Imager

Selecione então a fonte de coleta da imagem do disco como *Physical Drive* (o pendrive do suspeito) e clique em *Avançar*. Veja essa ação na **Figura 5**.

Escolha o pendrive conectado como drive de origem a partir do qual será feita a cópia bit a bit e clique em *Finish*. Em seguida clique em *Add...* para selecionar o diretório de saída, como mostra a **Figura 6**.

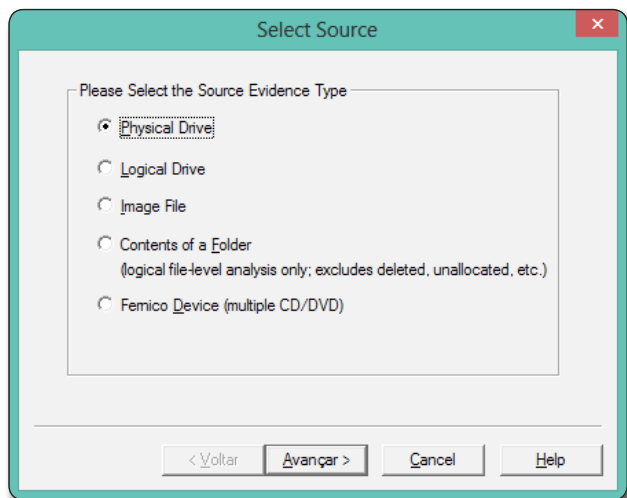


Figura 5. Fonte de origem para geração de cópia forense

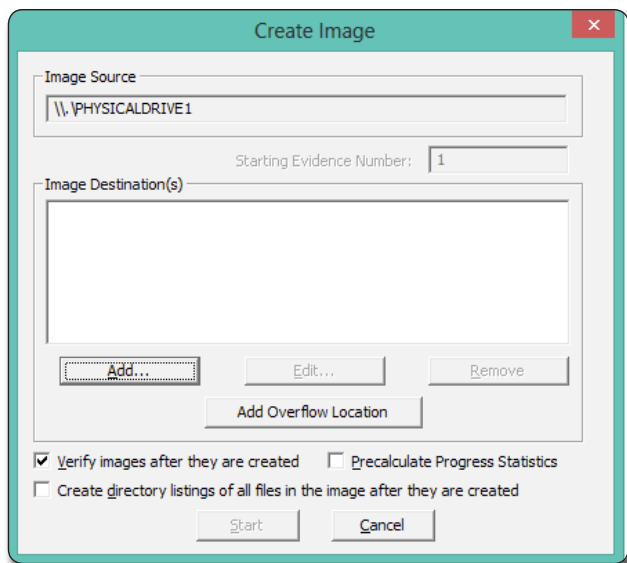


Figura 6. Personalizando o diretório de saída

Na escolha do formato de saída da cópia, devemos selecionar *raw* para obter a cópia bit a bit dos dados brutos do disco ou do volume escolhido, mantendo o conteúdo da origem fielmente replicado na cópia criada, e clicar novamente em *Avançar*. Este formato garantirá que as informações sejam copiadas por completo da origem para o destino designado.

Opcionalmente, preencha as informações do processo e registre as informações do caso e da evidência na próxima tela, e depois clique em *Avançar*.

Na tela exibida em seguida, será requisitado o diretório de destino aonde a cópia será criada. Devemos procurar e escolher no sistema de diretórios este local, especificar o nome do arquivo de imagem e definir o tamanho da fragmentação para 0 (zero), o que fará com que apenas um arquivo de imagem seja gerado em vez de muitos arquivos, de acordo com o tamanho do fragmento e o tamanho do disco ou volume envolvido no processo. Após definir esses três pontos, clicamos em *Finish* para avançar à próxima fase. Esta configuração pode ser visualizada na **Figura 7**.

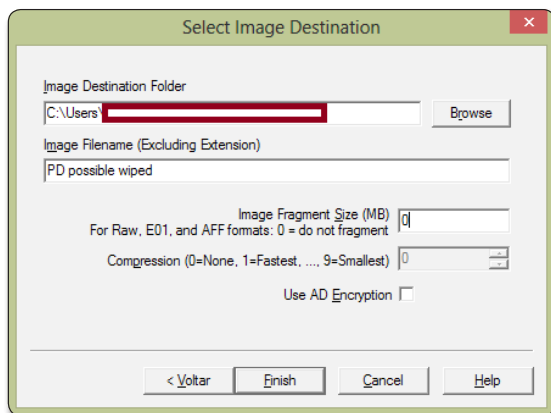


Figura 7. Escolhendo o diretório de destino e as configurações de fragmentação da imagem do sistema

Quando todas as opções estiverem prontas, clique em *Start Process* e aguarde o final do processo de duplicação do disco. A unidade será replicada preservando a sua integridade e mostrando o cálculo de hash no final (**Figura 8**). Se algo der errado, o hash será incompatível com a origem e você deverá analisar o que aconteceu para ocorrer o erro, como um erro de leitura ou problemas de alimentação elétrica. Após a criação do arquivo de imagem, será mostrado o processo de verificação de *hash*, aonde a origem da cópia e a cópia em si serão comparadas para atestar se a cópia do conteúdo foi fielmente executada ou se houve algum erro. Após o término da comparação, clique em *Close* nas duas janelas para continuar.

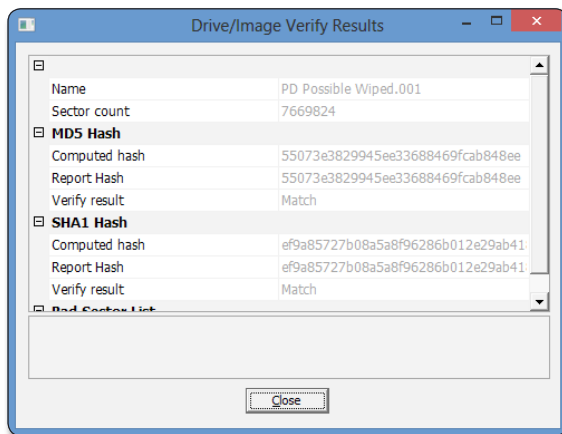


Figura 8. Resultados do processo de clonagem com a exibição das verificações de integridade

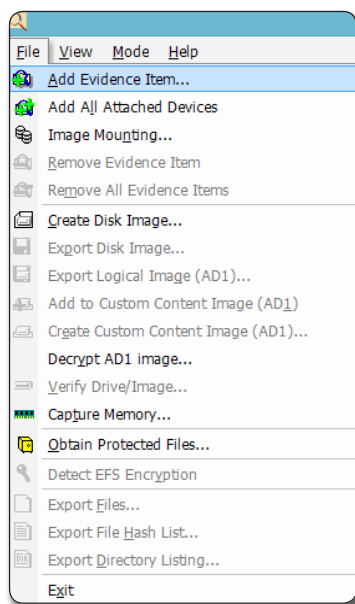


Figura 9. Menu do FTK Imager

Agora temos o arquivo de imagem do pendrive do suspeito e podemos começar a análise de seu conteúdo. Assim, utilizando o FTK Imager tentaremos descobrir e reconstituir fatos que possam apoiar nossa hipótese ou então fornecer subsídios para uma nova linha de investigação. Para carregar a cópia dentro do FTK Imager, utilize a opção *Add Evidence Item...*, exibida na Figura 9.

O processo de carregamento da cópia da evidência original no FTK Imager é semelhante à criação da imagem de disco a partir da origem, detalhada anteriormente neste artigo. Primeiro, escolhe-se o local de origem da cópia de disco que será montada no FTK Imager. Por exemplo, uma unidade física ou lógica (partição), um arquivo de imagem ou simplesmente uma pasta. Dentro das

opções disponíveis para montagem da evidência, vamos escolher a opção *Image File* e procurar no diretório onde se encontra o arquivo de imagem que criamos previamente. A estrutura da imagem pode ser visualizada na interface do FTK Imager, conforme a Figura 10.

Detalhando a listagem de arquivos no FTK Imager, podemos observar na Figura 11 que alguns arquivos possuem um marca em forma de X. Esta é a notação da ferramenta FTK Imager para informar que estes arquivos sofreram deleção no sistema de arquivos, no entanto ainda são passíveis de visualização quando a estrutura do volume é analisada profundamente pela ferramenta. Caso estes arquivos não tenham alguma sido sobrescritos no sistema de arquivos, poderão ser totalmente recuperados para nossa análise.

Podemos notar que muitos arquivos foram apagados do pendrive do suspeito. A recuperação destes pode nos ajudar a entender o motivo da exclusão destes dados. Para isso, selecione o arquivo que deseja recuperar com o botão direito do mouse e clique em *Export Files...*, como mostrado na Figura 12. Em seguida escolha um diretório para salvar o arquivo. Por fim, localize o arquivo no sistema de arquivos e visualize o seu conteúdo.

Após a exportação foi possível visualizar o conteúdo da apresentação “*Business Model - DRAFT 1.2 - with boss comments*” e perceber que se trata de um documento fundamental para uma nova área de negócio que a empresa pretende abrir para o mercado. Talvez um concorrente pudesse encontrar algumas informações valiosas dentro deste arquivo, comprometendo a competitividade e a concorrência leal entre as organizações.

Neste caso, a perícia é crucial como uma atividade preventiva e detectiva dentro da empresa. Com o auxílio de uma estrutura de monitoramento e uma equipe de colaboradores conscientizados, possíveis alertas podem ser gerados e investigações periciais conduzidas para que se evite um incidente de segurança de grande impacto para a empresa.

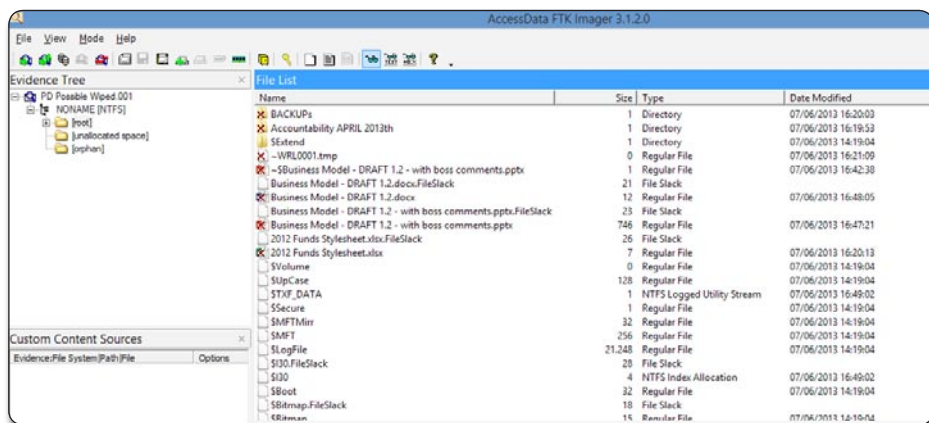


Figura 10. Conteúdo do sistema de arquivos do pendrive visualizado no FTK Imager

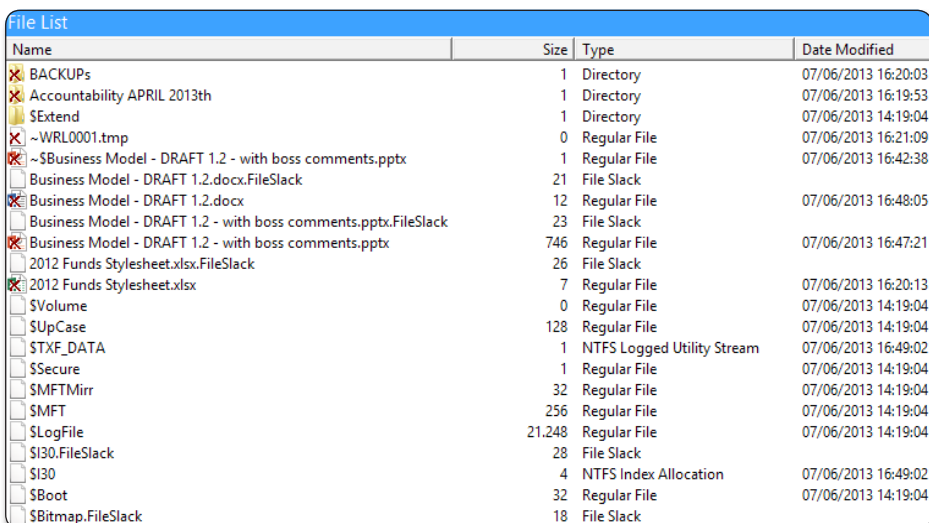


Figura 11. Visualização de arquivos deletados com uma marca em forma de X

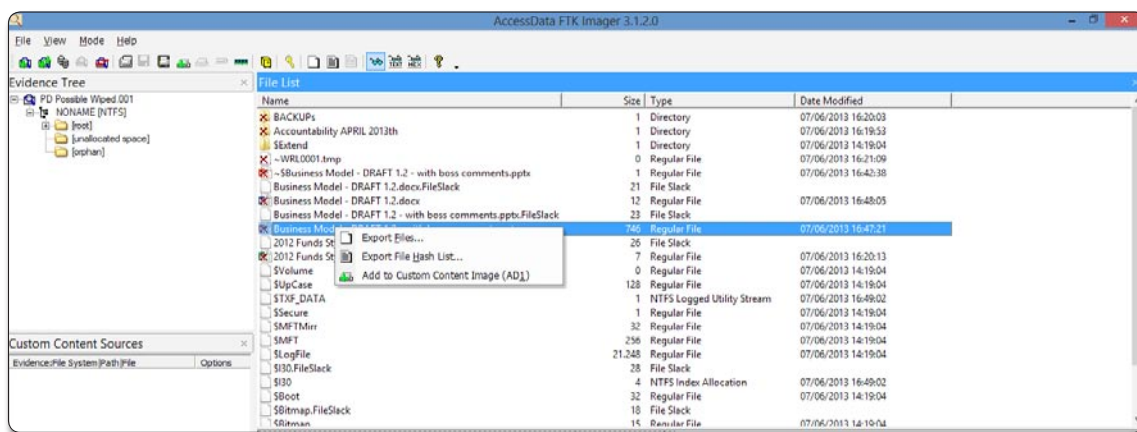


Figura 12. Exportação do conteúdo de um arquivo deletado

Conclusão

A partir deste artigo pudemos entender que o conhecimento quanto ao uso de ferramentas de análise forense é essencial para um perito, e vimos que o FTK Imager é uma ferramenta gratuita que oferece uma plataforma básica para investigações de evidências digitais.

Infelizmente a ferramenta não é dotada de um mecanismo de pesquisa e automação de busca, o que economizaria tempo ao facilitar a busca de conteúdos específicos, como arquivos PDF ou strings de texto, mas possui recursos que um investigador poder usar para iniciar suas atividades de coleta e análise de evidências que serão descritas posteriormente dentro de um detalhado laudo pericial.

Em muitas investigações, arquivos excluídos são um ponto-chave para mostrar o que aconteceu nos sistemas ou quais atividades são rotineiramente executadas no computador. Saber como apresentar esse tipo de evidência dentro do contexto de um caso pode ser essencial para encontrar respostas para o cliente, juiz ou tribunal que solicitou uma elucidação dos fatos que possibilite a atribuição de responsabilidades às pessoas envolvidas.

Para finalizar, é importante saber que o conhecimento necessário para se tornar um bom profissional em computação forense envolve diversas áreas de estudo, como sistemas operacionais, redes de computadores, segurança da informação e forense digital. Mesmo sendo uma área de difícil inserção profissional, a computação forense é um campo promissor em nosso país, está recheada de desafios e carece de profissionais para atender à crescente demanda investigativa em consultorias e grandes corporações.

Autor



Nichols Jasper

É especialista em segurança da informação com experiência em serviços de consultoria nas áreas de forense computacional e investigação de incidentes de segurança, envolvendo casos de fraude corporativa com violação de propriedade intelectual e modificação fraudulenta de registros eletrônicos.

Links:

Forensics Duplicator

<http://www.tableau.com/index.php?pageid=products&category=duplicators>

Forensics: What happens when files are deleted?

<http://whereismydata.wordpress.com/2009/05/02/forensics-what-happens-when-files-are-deleted/> –

Bloqueadores de Escrita de disco

http://www.forensicswiki.org/wiki/Write_Blockers

CAINE Computer Forensics Linux Live Distro

<http://www.caine-live.net/>

Informações sobre a Ferramenta FTK Imager

http://www.forensicswiki.org/wiki/FTK_Imager

RFC 3227 Guidelines for Evidence Collection and Archiving

<http://www.ietf.org/rfc/rfc3227.txt>

Biblioteca Acadêmica Data Security

<http://www.datasecurity.com.br/index.php/biblioteca-data-security>

Formação em Computação Forense

<http://www.datasecurity.com.br/index.php/cursos/forense-computacional>

Endereço para download do FTK Imager

<http://www.accessdata.com/support/product-downloads>

Autor



Marcelo Lau

marcelo.lau@datasecurity.com.br

Engenheiro, pós-graduado em Administração e Comunicação e Arte. Possui um mestrado pela Universidade de São Paulo e possui experiência em Segurança da Informação e Computação Forense em diversos grandes bancos do Brasil. Atualmente é o proprietário e diretor executivo da Data Security no Brasil. Leciona aulas como professor em diversas universidades do Brasil e cursos em países da América Latina, como Argentina, Bolívia, Colômbia, Paraguai e Peru.

Desmistificando o Storage Corporativo

Entenda todas essas siglas e o que está por trás delas

De uma década para cá, o storage corporativo se tornou cada vez mais presente na infraestrutura de TI das empresas. É um equipamento fundamental para quem deseja disponibilidade e alto desempenho. O problema é que, por se tratar de uma tecnologia relativamente recente, e que poucos profissionais, particularmente em início de carreira, têm a oportunidade de conhecer de perto, acaba se criando um mito em torno do assunto, o que só contribui para a falta de informação. E porque é importante conhecer o storage? Porque o volume de dados vem crescendo em um ritmo vertiginoso, e nada indica que essa tendência vá mudar nos próximos anos. Para lidar com volumes de dados multi-terabyte, o armazenamento local (disco interno) simplesmente não é uma opção. Fora isso, conforme o volume de dados aumenta, em muitos casos a demanda por desempenho aumenta, logo é necessário um equipamento que tenha os recursos e a robustez necessária para suportar essa demanda.

Com base em tudo o que foi citado, este artigo se propõe a desmistificar o storage ao abordar os principais assuntos relacionados de forma clara e objetiva. A ideia não é que o leitor se transforme em um especialista em storage da noite para o dia, mas sim que o conteúdo exposto aqui sirva de ponto de partida para um aprendizado mais profundo, sem deixar de lado a necessidade do saber prático para o dia a dia.

Configuração Física

Neste tópico mostraremos os componentes de hardware mais comuns em uma solução de storage, bem como um exemplo de arquitetura típica de SAN.

Hardware

Como usuários finais de um storage corporativo, vemos apenas os discos acessíveis por um servidor. Para nós e para o servidor é transparente se esse disco é local ou se faz parte de uma SAN (Storage Area Network). A seguir, descrevemos os equipamentos necessários para que uma solução desse tipo funcione corretamente.

Resumo DevMan

Porque este artigo é útil:

O objetivo deste artigo é desmistificar o storage corporativo para os profissionais que não tiveram oportunidade de ver de perto ou trabalhar com estes equipamentos, mas administram aplicações que dependem destes, como bancos de dados, servidores de e-mail corporativo e servidores de arquivo, por exemplo. É muito fácil se perder na sopa de letrinhas do mundo do storage (SAN, FC, SAS, LUN, RAID, GBIC, iSCSI), mas esta breve explicação deve fornecer uma base teórica razoável para que qualquer não especialista seja capaz de entender do assunto, ao ponto de poder discutir com um especialista em storage para explicar uma demanda corretamente.

Conhecer a terminologia e os conceitos envolvidos em soluções de storage é essencial para todo profissional que administra aplicações que delas dependem (bancos de dados, e-mail, servidor de arquivos, dentre outras).

Storage

O storage é um equipamento que permite a instalação de uma grande quantidade de discos de tipos diversos (veja o **BOX 1**) e possui formas de conectividade para permitir o acesso por diversos servidores. O storage é tipicamente composto por:

- **Controladoras ou storage processors (Figura 1)**, que gerenciam toda a configuração do equipamento, a conectividade externa, o acesso aos discos e ao cache de dados. Normalmente há mais de uma, para redundância em caso de falha;
- **Gavetas ou Drive enclosures (Figura 2)**, que são os acessórios que recebem os discos.

Os equipamentos mais sofisticados são construídos de forma modular, de modo que é possível a expansão pela adição de mais controladoras ou drive enclosures posteriormente. Na **Figura 3** é possível ver um storage completo com seus módulos montado em rack.

Switch

Os switches (**Figura 4**) têm o papel de implementar a conectividade entre os servidores e o storage. Normalmente é utilizado mais de um, também por questões de redundância.



Figura 1. Storage Processor



Figura 2. Drive Enclosure



Figura 3. Storage montado em rack, visão frontal



Figura 4. Switch SAN. Repare nas portas vazias, prontas para receber as GBICs

A conexão com os servidores é feita a partir de cabos, cujo tipo varia de acordo com o switch utilizado. Os tipos mais comuns atualmente são:

- **Fiber Channel (FC)**, que utiliza fibra ótica para o tráfego de dados, e que suporta desde distâncias pequenas (até 100m, chamada “shortwave”) até grandes distâncias (até 50 km, chamada “longwave”). Conexões FC utilizam um tipo de switch especial comumente chamado de “SAN switch”;
- **iSCSI (Internet SCSI)**, que utiliza switches de rede comuns e cabos de rede UTP (par trançado) ou fibra ótica.

BOX 1. Tipos de Disco

Os equipamentos atuais suportam o que há de mais moderno em tecnologia de disco. Internamente os discos de tecnologia tradicional (placas giratórias com cabeças de leitura) são bastante similares. Eles diferem no tipo de conexão e na velocidade de rotação. Quanto maior a rotação, maior o desempenho. A seguir, temos uma breve descrição dos tipos mais comuns:

- **FC (Fiber Channel)**: São discos de alto desempenho, com opções de 10.000 e 15.000 RPM, muito utilizados em equipamentos de grande porte. Atualmente são oferecidos em tamanhos diversos, que podem variar de 146 GB a 2 TB;
- **SAS (Serial Attached SCSI)**: São discos de alto desempenho que utilizam a conexão SAS de 6 Gbps, com opções de 10.000 e 15.000 RPM, muito utilizados em equipamentos de pequeno a grande porte. Atualmente são oferecidos em tamanhos de 300, 450 e 600 GB;
- **NL-SAS (Near-Line SAS)**: São discos de baixo custo que utilizam a conexão SAS. Normalmente com velocidade de 7.200 RPM, são utilizados para dados que não possuem alta demanda de acesso, como dados históricos, por exemplo. Atualmente são oferecidos em tamanhos de 1 a 3 TB;
- **SATA (Serial ATA)**: São discos de baixo custo que utilizam a conexão SATA (Serial ATA) de 3 Gbps. São encontrados em equipamentos um pouco mais antigos e possuem função similar à dos discos NL-SAS;
- **SSD (Solid State Disk)**: São discos baseados em memória flash, que oferecem o melhor desempenho entre todos os tipos mencionados acima, por não possuírem partes móveis. É uma tecnologia recente e bastante promissora, e por isso, ainda muito cara para uso em larga escala. São utilizados em aplicações onde o máximo desempenho é essencial.

As conexões FC podem operar em velocidades que variam de 1 a 16 Gbps, sendo que as mais comuns atualmente são de 4 e 8 Gbps. As conexões iSCSI dependem da infraestrutura de rede, e podem utilizar conexões de 1 ou 10 Gbps, dependendo do switch de rede utilizado.

O fato de a tecnologia iSCSI utilizar switches de rede comuns traz uma grande vantagem em termos de custo, pelo fato desta infraestrutura já existir nas empresas, mas normalmente com desempenho inferior aos switches FC (Fiber Channel), devido ao fato de trafegar comandos SCSI sobre o protocolo TCP/IP, o que tem um impacto significativo. Switches FC é a escolha comum em soluções que demandam alto desempenho.

As conexões FC e iSCSI de 10 Gbps, que utilizam fibras óticas como meio, necessitam de transceivers (conversores de mídia) para cada porta de conexão. Esses conversores (veja a **Figura 5**) se chamam GBIC (*Gigabit Interface Converter*), e servem para converter o sinal da fibra (luz) para o sinal eletrônico do switch e vice-versa. Na **Figura 6** é possível ver um exemplo de conector de fibra ótica, utilizado para conexão com GBICs. É comum se utilizar o termo “fabric” para descrever o switch SAN.



Figura 5. GBIC – Gigabit Interface Converter

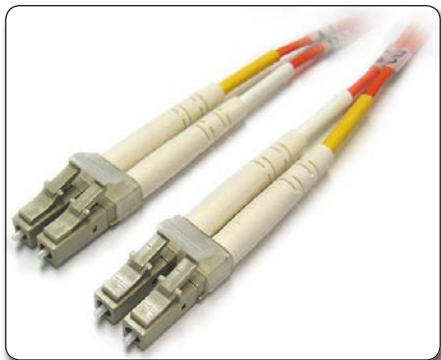


Figura 6. Conector de fibra ótica

HBA (Host Bus Adapter)

HBA (Figura 7) é a placa que instalamos nos servidores para permitir a conectividade com a SAN. Normalmente possuem duas portas (para redundância) e existem modelos específicos para conexões FC e iSCSI (no caso do iSCSI, podem ser utilizadas até placas de rede comuns).

A HBA utilizada em conexões FC possui um identificador único chamado WWN (*World Wide Name*), que possui papel similar a um endereço MAC de uma placa de rede. Esse identificador é utilizado nas configurações realizadas nos switches SAN para identificar as conexões de cada servidor.



Figura 7. HBA – Host Bus Adapter

Todos esses equipamentos se comunicam utilizando os comandos do protocolo SCSI (*Small Computer System Interface*), criado nos anos 80 e que originou diversos conectores, que começaram a cair em desuso com a chegada do SAS (que significa *Serial Attached SCSI*, uma evolução dos padrões anteriores). Os conectores SCSI se foram, mas o protocolo permaneceu.

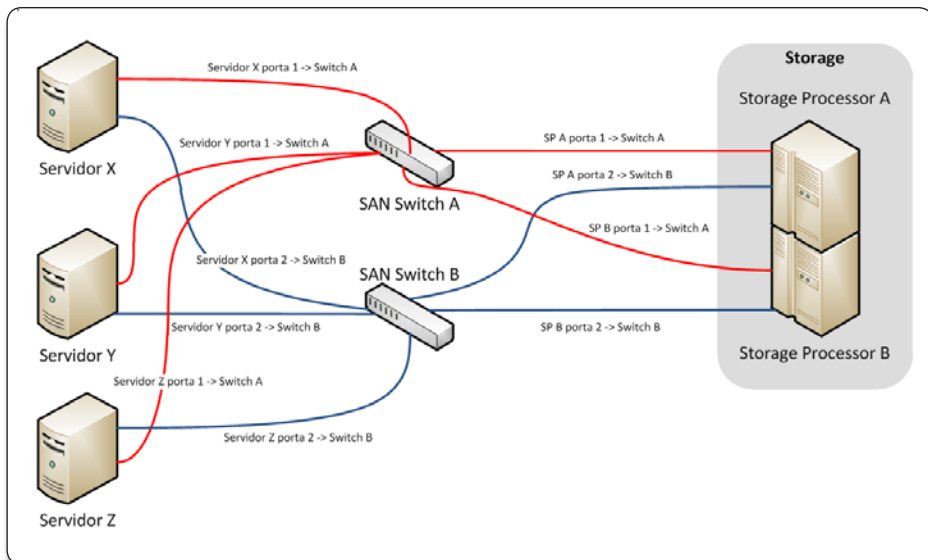


Figura 8. Arquitetura Típica de uma SAN

Arquitetura Física Típica

Um disco tradicional funciona dentro do servidor, com distâncias de cabos pequenas (10-20 cm) e com cabeamento e conectores protegidos pelo gabinete. Quando falamos de SAN, os equipamentos muitas vezes estão em racks separados, e às vezes até em prédios separados, o que traz um fator de risco adicional à solução. Por esse motivo a redundância é fundamental em uma solução de storage.

No diagrama apresentado na Figura 8 podemos ver uma arquitetura típica de SAN: um storage com duas controladoras que possuem duas portas cada, dois switches SAN e alguns servidores. Cada controladora passa por cada um dos switches, e cada porta das placas HBA dos servidores também vão para switches separados. Dessa forma, se um switch, uma controladora ou uma porta de HBA falhar, o sistema inteiro continua a funcionar.

Configuração lógica

Depois de explicar os componentes físicos da solução, vamos à configuração lógica dos equipamentos.

Uma vez que o storage esteja instalado (processo que normalmente é feito pelo fabricante), os servidores com suas placas HBA, e tudo isso conectado através de switches SAN, como no desenho da Figura 8, temos duas tarefas importantes:

- **Configuração dos discos:** é a configuração dos discos para permitir que sejam utilizados pelas aplicações;
- **Configuração da conectividade:** é a configuração que permite que todos os equipamentos se comuniquem entre si, e mais importante, que não acessem recursos indevidamente.

Configuração dos discos – O que é RAID ?

Quando pensamos em storage corporativo, 100% das vezes pensamos em RAID. O RAID traz diversas vantagens em relação à configuração de um disco stand alone, como por exemplo, segurança e melhor desempenho. Aqui vale uma pausa para discutirmos o que é RAID. O termo RAID significa Redundant Array of Independent Disks, e é uma tecnologia que permite o agrupamento lógico de discos (array) para obter melhor desempenho, segurança ou custo. O RAID é classificado em níveis (levels), que determinam o funcionamento dos discos na configuração selecionada, e o tipo de redundância.

Para determinar o nível de RAID ideal para cada caso sempre devem levar em conta três fatores: custo, desempenho e segurança de dados. Normalmente quando obtém o melhor em dois desses fatores, se perde no terceiro. É tudo uma questão de escolher o mais apropriado para cada necessidade.

Existem vários níveis de RAID, mas os que foram consagrados pelo uso e, portanto são os mais comuns, são apresentados a seguir:

- **0 (striping):** É o nível de RAID que oferece o melhor desempenho e custo, com a contrapartida de não oferecer segurança. Os dados são distribuídos pouco a pouco em cada disco do array, o que faz com que o acesso de leitura e escrita seja muito rápido, mas em caso de perda de um disco, o array inteiro falha. Só é utilizado em situações particulares onde a segurança de dados não é importante (por exemplo, discos temporários para renderização de vídeos, que demandam alto desempenho, mas que o processo pode ser reiniciado caso os discos falhem);

- **1 (mirroring - espelhamento):** Oferece ótima relação entre desempenho e segurança, com o custo de utilizar no mínimo metade do espaço total para garantir a redundância. Todos os dados que são gravados em um disco são automaticamente gravados em um ou mais discos, o que garante a segurança dos dados caso um disco falhe. Teoricamente, no entanto, é possível utilizar mais de dois discos em RAID 1, o que aumentaria o número de cópias dos dados (um array RAID 1 com três discos, por exemplo, possui o dado original em um disco mais duas cópias), mas na prática isso não é comum. Normalmente são utilizados apenas dois discos para formar um array RAID 1, que tolera a perda de apenas um disco;

- **10 (striping de mirror):** Funciona como um array RAID 0 construído no topo de dois ou mais arrays RAID 1. Com isso oferece desempenho próximo ao do RAID 0 com a redundância do RAID 1. É o nível mais indicado quando se deseja máximo desempenho com segurança, mas é o mais caro de todos: utiliza metade do espaço total para redundância, como no RAID 1, e são necessários ao menos quatro discos para formar um array RAID 10.

Para este nível de RAID, o mecanismo de tolerância a falhas é um pouco mais complicado. Um array RAID 10 tolera a perda de até um disco por “array” RAID 1, ou seja, se temos seis discos em RAID 10, podemos pensar em três pares de discos em RAID 1 unidos para formar um array RAID 0. Nesse caso, podemos perder até um disco de cada par, mas não os dois discos de um mesmo par;

- **5 (striping com paridade distribuída):** Oferece ótima relação entre custo e segurança, mas o desempenho de escrita é abaixo dos demais devido ao esforço de manutenção dos dados de paridade, que servem para reconstruir o array no caso de falha de um disco. Utiliza normalmente o espaço de um disco apenas para manter a redundância, e são necessários no mínimo três discos para formar um array RAID 5. Tolerar a perda de até um disco.

O cálculo de paridade, que garante a segurança do RAID 5, funciona da seguinte forma: pense em três discos idênticos, onde todas as posições de gravação de dados possuem um endereço específico que seja comum a todos os discos, ou seja, a terceira posição de todos os discos se chama “3”, por exemplo. Ao gravar um bit em um determinado endereço, a controladora RAID faz um cálculo (XOR, operador lógico “ou exclusivo”) entre esse bit e o bit do endereço três de um dos discos vizinhos, gerando um terceiro

bit, que é o bit de paridade. Esse bit de paridade será gravado no endereço três de um disco diferente dos dois utilizados para o cálculo. Isso é realizado para todos os dados que são gravados em um array RAID 5, e esse processo é feito de uma forma que a cada dado gravado, a combinação de discos utilizados para gerar a paridade e o disco utilizado para gravá-la sejam diferentes. Dessa forma, quando um disco é perdido, é possível reconstruir os seus dados a partir dos dados e bits de paridade dos demais;

- **6 (striping com dupla paridade):** Similar ao RAID 5, mas grava dois bits de paridade para cada bit gravado. Por isso tem desempenho de escrita fraco, mas tem a grande vantagem de suportar a falha de até dois discos do array.

Para concluir, vale notar que boa parte dos equipamentos atuais de storage suporta a configuração de um ou mais discos como hot-spare, que têm a função de “estepe” em situações de falha. Nesses casos, o storage, ao detectar uma falha, pode iniciar um rebuild do array utilizando o disco de spare automaticamente, sem intervenção humana.

Seleção do nível de RAID

Para determinar qual nível de RAID devemos utilizar, é preciso pensar em quatro questões importantes:

- Quanto espaço é necessário?
- Qual é o orçamento?
- Qual é a necessidade de segurança?
- Qual é a necessidade de desempenho?

Na prática, essas perguntas devem vir antes da compra dos equipamentos, para determinar a configuração do equipamento a ser comprado. No entanto, para o nosso exercício mental de não especialistas, vale fazê-las agora, supondo que temos ao nosso dispor todos os recursos que desejarmos (a realidade normalmente é bem diferente).

Conhecer bem os níveis de RAID, ao ponto de sugerir com autoridade qual deve ser utilizado, é fundamental para todo profissional de infraestrutura, até para evitar generalizações do tipo “RAID 5 é ruim” ou “RAID 10 é o melhor”. Cada um é mais apropriado para determinado caso.

Para responder às três primeiras perguntas, não é preciso um grande esforço, e elas estão intimamente ligadas. No entanto, a resposta à questão do desempenho é a mais complicada e que exige mais atenção. Na seção final deste artigo, que fala de workloads e IOPS, podemos ver uma explicação detalhada sobre a relação entre workload, IOPS e nível RAID, bem como exemplos dessa lógica.

Uma vez determinado o nível RAID, podemos criar nosso primeiro array de discos, também conhecido como RAID group. Esse array deve ser dividido em Logical Units, ou LUNs (LUN no singular). A logical unit é como uma partição de disco, só que de fato é a partição de um array, que por sua vez é a união de vários discos (partição da união!?). Simplificando: os discos são unidos fisicamente como um array com a configuração de nível RAID, e depois divididos logicamente em LUNs para serem acessados

pelos servidores. Uma vez que a LUN seja apresentada ao servidor, este passa a tratá-la como um disco comum, que deve ser formatado para depois ser utilizado.

Configuração da conectividade

Para que o disco possa ser apresentado ao servidor, existem mais alguns passos a serem realizados, que estão relacionados à segurança:

- **Zoning:** é a configuração feita no switch SAN que diz quais portas falam com quais. Normalmente é feita de forma que a HBA de um servidor acesse somente a HBA da controladora do storage, de modo a não permitir acessos indevidos;
- **Configuração de Storage Groups:** é a configuração feita no storage que diz quais LUNs podem ser acessadas por quais servidores. É a segunda camada de proteção a acessos indevidos, ao não permitir que um servidor acesse um disco que não lhe pertence. Aqui vale um parêntese: em ambientes de cluster, onde dois ou mais servidores precisam acessar um determinado disco, todos os servidores devem fazer parte do mesmo storage group que os discos, e o controle de qual servidor acessa qual disco é feito via software.

O ato de adicionar uma LUN a um storage group existente configura o ato de apresentar o disco para o servidor. A partir deste momento, o disco estará visível para o sistema operacional.

Concluídas estas tarefas, o storage está pronto para ser utilizado.

Funcionalidades importantes

Após ler a explicação até aqui, deve estar claro que o storage oferece uma série de vantagens em relação a uma configuração de disco interno, principalmente na capacidade de implementar soluções de alto desempenho.

Além disso, o storage normalmente possui uma série de funcionalidades adicionais que são menos conhecidas, mas não menos importantes, do ponto de vista de gerenciamento. A seguir estão descritas algumas dessas funcionalidades:

- **Clone:** é a capacidade de clonar os dados de uma determinada LUN. A vantagem do clone em comparação a uma cópia de arquivos do sistema operacional, é que o clone ocorre inteiro dentro do storage, sem tráfego pela SAN, e por isso tende a ser muito mais rápido. É útil em migrações de servidores onde se deseja ter um backup completo e rápido;
- **Snapshot:** é a capacidade de tirar uma “fotografia” de uma determinada LUN em uma posição no tempo. Snapshots normalmente são implementados utilizando uma técnica chamada “copy-on-write”, que somente grava a versão antiga dos dados no snapshot conforme estes são alterados na LUN original. Dessa forma, o snapshot tem um tamanho bastante pequeno quando criado, e vai crescendo conforme a LUN original é alterada. A função mais importante do snapshot é permitir que se restaure o estado dos dados de uma LUN àquele que existia no momento em que o snapshot foi criado. É útil em aplicações de patches, quando se deseja poder desfazer as alterações sem restaurar um

backup, por exemplo;

- **Thin provisioning:** é a capacidade de alocar uma LUN que possua alocação dinâmica, ou seja, podemos criar uma LUN “thin” de 100 GB, mas ela só ocupará o espaço que estiver em uso pelo servidor, similar aos discos de crescimento dinâmico dos produtos de virtualização. Isso é útil quando não temos uma estimativa precisa da necessidade de alocação de espaço para um servidor, e, portanto não queremos comprometer uma área do storage desnecessariamente;

- **Storage Pools e Tiering:** é a alternativa aos arrays/RAID groups tradicionais, e que vem ganhando espaço ultimamente. Em vez de configurar arrays para cada aplicação, selecionando um tipo de disco específico, alguns equipamentos oferecem a opção de criar um storage pool com muitos, se não todos os discos do storage, misturando discos rápidos e lentos. O storage passa então a analisar a carga submetida e a identificar o perfil de acesso de cada dado. Uma vez identificado o perfil, o storage realoca dados muito acessados para discos mais rápidos, e dados pouco acessados para discos mais lentos.

As LUNs são criadas diretamente sobre o storage pool, e você pode recomendar (repare o termo) em qual camada (“tier”) de discos aquela LUN deve operar. No final, quem decide onde a LUN vai ficar é o storage, após a análise, que tipicamente ocorre uma vez por dia. A ideia aqui é que o storage tem mais condições do que nós de analisar os dados de acesso para decidir onde é melhor colocar cada LUN, e ele tem a capacidade de refazer essa alocação todos os dias. Na prática, nem os fabricantes recomendam o uso de pools para aplicações que exigem máximo desempenho, mas para ambientes onde o desempenho não é tão crítico, a facilidade de administração é bastante compensadora, pois libera o administrador da tarefa de controlar a ocupação array por array;

- **Replicação block-level:** é a capacidade de manter uma replicação online de LUNs no mesmo storage ou em storages diferentes, até em localidades diferentes, no nível de bloco do disco, sem se importar com o conteúdo. A LUN replicada fica em um estado chamado de “crash-consistent”, ou seja, similar ao estado em que um disco fica quando o computador é desligado abruptamente. Contudo, com os file systems modernos (NTFS, ext3, ext4) que possuem journaling, isso normalmente não é um problema. Essa funcionalidade é muito útil para a criação de sites de contingência (disaster recovery). Soluções de replicação block-level normalmente precisam de um software opcional instalado no storage, comprado à parte.

Workloads, IOPS, eficiência de espaço e “calculadora RAID”

Normalmente utilizamos a métrica de IOPS (I/Os por segundo) para falar de desempenho, ou mais especificamente de carga de trabalho (workload), quando vamos especificar a necessidade de desempenho de um array de discos para uma determinada aplicação, e a partir dessa informação realizar o dimensionamento apropriado. Cada tipo de disco possui um valor nominal de IOPS suportado, que somado ao tipo de RAID e à quantidade de discos do array nos dá uma estimativa de quantos IOPS o array irá suportar.

Na **Tabela 1** podemos ver uma lista de valores típicos de IOPS para modelos comuns de disco, retirada do manual de um equipamento de porte intermediário de um grande fabricante do mercado (considerando I/O randômico, pior caso).

Tipo de Disco	IOPS (médio)
NL-SAS e SATA 7.200 RPM	90
SAS e FC 10.000 RPM	140
SAS e FC 15.000 RPM	180
SSD	3500

Tabela 1. Valores médios de IOPS por tipo de disco

Cada nível de RAID, devido às suas particularidades de implementação, possui determinado fator de impacto de escrita. O fator de impacto funciona como um multiplicador, de modo que para obtermos a quantidade real de escritas que serão realizadas no array, é preciso multiplicar a quantidade de escritas realizadas pelo sistema operacional pelo fator de impacto.

Quanto maior o fator de impacto, menor o desempenho de escrita no array. A **Tabela 2** mostra os fatores de impacto para os níveis de RAID mais comuns.

Nível RAID	Fator de Impacto na Escrita
1	2
5	4
6	6
10	2

Tabela 2. Fatores de Impacto de Escrita por nível de RAID

Quando dizemos que o RAID 5 possui impacto 4, isso significa que para cada escrita realizada por uma aplicação no servidor, o storage irá realizar 4 escritas nos discos que compõem o array.

Outra informação importante é a eficiência de espaço de cada nível de RAID. Precisamos desses valores para determinar qual será o espaço útil do nosso array. Na **Tabela 3** podemos ver as fórmulas de eficiência de espaço para os níveis de RAID mais comuns.

Nível RAID	Eficiência	Exemplo
1	x/2	2 discos de 500 GB em RAID 1 geram 500 GB úteis (1000 GB/2)
5	1-(1/n)	3 discos de 500 GB em RAID 5 geram 1 TB útil (1500 GB – 1500 GB/3 discos)
6	1-(2/n)	6 discos de 500 GB em RAID 6 geram 2 TB úteis (3000 GB – 6000 GB/6 discos)
10	x/2	4 discos de 500 GB em RAID 10 geram 1 TB útil (2000 GB/2)

Tabela 3. Eficiência de Espaço por nível de RAID

Resumindo, para determinar a configuração de um array é preciso considerar:

- Capacidade de IOPS do tipo de disco selecionado;

- Distribuição do I/O entre leituras e escritas (percentual de cada), para avaliar o impacto da escolha do nível do RAID;
- Eficiência do nível do RAID escolhido, para determinar quanto espaço útil terá disponível.

Outra informação importante é o tipo de I/O realizado pela aplicação, se sequencial ou randômico. No nosso caso, como estamos utilizando valores de IOPS pensando em acesso randômico, que é o pior caso, podemos ignorar essa informação (o arquiteto de storage normalmente não pode, mas nós não somos os especialistas aqui).

Tomando como base os números das **Tabelas 1, 2 e 3**, podemos pensar em uma série de cálculos para nos guiar no dimensionamento de arrays.

Para exercitar os conceitos apresentados, propomos dois exemplos aqui, que demonstram bem a complexidade do assunto storage, e que servirão como base para o cálculo de dimensionamento.

Exemplo 1: Dimensione um array de discos em um storage para suportar um banco de dados de 5 TB. Temos a informação de que é um banco de dados com grande volume de transações durante o dia, que em momentos de pico atinge 5.000 IOPS, e que 60% desse acesso é de escrita.

Ao analisar o enunciado do problema, podemos perceber que:

- 60% do acesso é de escrita. Assim sendo, não faz sentido escolhermos um nível de RAID que penalize muito a escrita, como os níveis 5 ou 6. Por se tratar de um banco de dados, podemos supor que há a necessidade de redundância, o que faz com que descartemos o nível 0. Nossa escolha nesse caso será pelo RAID 10;

- Comparando a demanda de IOPS (5.000) com o volume de IOPS médio suportado por um disco rápido, não-SSD (180, de acordo com a **Tabela 1**), podemos perceber que precisaremos de muitos discos:

$$5.000 \text{ IOPS} / 180 \text{ IOPS por disco} = 27,7 \text{ discos}$$

- Ao considerarmos que 60% desse acesso é de escrita, e o fator de impacto de escrita do RAID 10 é 2, de acordo com a **Tabela 2**, temos o seguinte cálculo:

$$((5000 * 60\%) * 2) + (5000 * 40\%) / 180 = 44,4 \text{ discos}$$

Decompondo a fórmula, temos que:

$$60\% \text{ de } 5000 \text{ IOPS de escrita} = 3000 \text{ IOPS}$$

$$40\% \text{ de } 5000 \text{ IOPS de leitura} = 2000 \text{ IOPS}$$

$$3000 \text{ IOPS de escrita} * 2 \text{ (fator de impacto de escrita do RAID 10)} = 6000 \text{ IOPS}$$

$$\text{Total de IOPS} = 6000 + 2000 = 8000 \text{ IOPS}$$

$$8000 \text{ (Total de IOPS)} / 180 \text{ (IOPS por disco)} = 44,4 \text{ discos}$$

Desse exemplo, podemos derivar a seguinte fórmula:

$$D = ((iops * w) * i) + (iops * r) / iops_avg$$

Onde:

D: Total de discos a ser utilizado;

iops: Total de IOPS para o qual se deseja dimensionar o array;

i: fator de impacto de escrita para o nível de RAID;

w: percentual de escrita dos IOPS totais;

r: percentual de leitura dos IOPS totais;

iops_avg: IOPS médio do tipo de disco a ser utilizado no array;

Após calcular a quantidade de discos necessária, é preciso definir o tamanho do disco a ser utilizado. Ao consultar a **Tabela 3** podemos obter a fórmula de eficiência de espaço do RAID 10, que é $x/2$, onde x é a capacidade somada de todos os discos. Pelo cálculo anterior, identificamos que precisamos de 5 TB em 44 discos. Logo, temos:

$$5000 = 44x/2$$

ou

$$x = 10000/44$$

$$x = 227 \text{ GB por disco}$$

Arredondando o resultado obtido para o próximo tamanho comum de disco, temos discos de **300 GB**. Para confirmarmos se o nosso cálculo está correto, podemos inverter a conta, e com o tamanho de disco proposto, temos:

$$(300 \times 44) / 2 = 6600 \text{ GB}$$

Assim, com 44 discos FC ou SAS de 15.000 RPM e 300 GB, conseguimos atender à demanda do nosso banco de dados com folga.

Exemplo 2: Dimensione um array de discos em um storage para suportar um servidor de arquivos de 10 TB. Temos a informação de que é um servidor de arquivos pouco utilizado, com muitos dados históricos, que em momentos de pico atinge 500 IOPS, e que 80% desse acesso são de leitura.

Ao analisar o enunciado do problema, podemos perceber que:

- 80% do acesso são de leitura e precisamos de 10 TB úteis. Nesse caso, podemos escolher um nível de RAID mais econômico, como o 5 ou 6. Para efeito de comparação, faremos a conta com os dois;
- Se o servidor de arquivos é pouco utilizado, podemos configurar o nosso array com discos mais lentos e baratos. Escolhemos, então, os discos NL-SAS de 7200 RPM, que fazem 90 IOPS cada.

A seguir é apresentado o exemplo de cálculo para um RAID 5 com discos NL-SAS de 7200 RPM:

Quantidade de IOPS médio do disco NL-SAS 7200 RPM (**Tabela 1**): 90

Fator de impacto do RAID 5 (**Tabela 2**): 4

Eficiência de espaço do RAID 5 (**Tabela 3**): $1-1/n$

Repetindo a fórmula do exemplo anterior, temos:

$$((500 \times 20\%) \times 4) + (500 \times 80\%) / 90 = 8,88 \text{ discos}$$

Arredondando para o inteiro mais próximo, temos **nove discos** em RAID 5.

Para determinar o tamanho do disco desejado, utilizamos a fórmula de eficiência do RAID 5:

$$1-(1/9) = 0,89$$

10 TB (total de espaço necessário) / 9 discos / 0,89 = **1,25 TB por disco**

Agora, faremos o cálculo considerando o uso de RAID 6 com discos NL-SAS de 7200 RPM:

Quantidade de IOPS médio do disco NL-SAS 7200 RPM (**Tabela 1**): 90

Fator de impacto do RAID 6 (**Tabela 2**): 6

Eficiência de espaço do RAID 6 (**Tabela 3**): $1-2/n$

Repetindo a fórmula do exemplo anterior, temos:

$$((500 \times 20\%) \times 6) + (500 \times 80\%) / 90 = 11,11$$

Não perca tempo reinventando a roda!

COBREBEMX

Componente completo para sua Cobrança por Boleto Bancário e Débito em Conta Corrente

Mais de 40 exemplos em diversas linguagens de programação

Geração e leitura de arquivos (remessa e retorno) nos padrões FEBRABAN e CNAB

Testes e Downloads gratuitos em nosso site

ACESSE E CONHEÇA O COMPONENTE EM:
WWW.COBBREBEM.COM

Arredondando para o inteiro mais próximo, temos **11 discos** em RAID 6.

Para determinar o tamanho do disco desejado, utilizamos a fórmula de eficiência:

$$1 - (2/11) = 0,81$$

10 TB (total de espaço necessário) / 11 discos / 0,81 = **1,12 TB por disco**

Nos dois casos, arredondando o resultado para o próximo tamanho de disco existente no mercado, temos discos de 1,5 TB. Se optarmos pelo RAID 6, serão necessários dois discos a mais, porém com suporte à falha de até dois discos, o que pode ser uma vantagem dependendo da importância dos dados em questão.

Conclusão

Neste artigo foram apresentados os principais componentes envolvidos em uma solução de storage corporativo, as principais configurações necessárias e foi fornecida uma explicação sobre RAID e suas indicações. Além disso, foram mostrados exemplos de workload e de dimensionamento para volume e desempenho.

É importante reforçar que sempre devemos confiar o trabalho de configuração ao arquiteto ou administrador do storage, que tem a experiência para avaliar todos os parâmetros necessários e tomar a melhor decisão possível. Estes profissionais normalmente precisam levar em consideração outros fatores ao desenhar uma solução de storage, como o número ótimo de discos por array dependendo do nível de RAID e a quantidade de discos disponível. A questão é que, apesar de ser o especialista em storage, nem sempre esse técnico é especialista na sua aplicação, e aí é importante saber falar a “língua” do storage para conseguir estabelecer essa ponte e obter o melhor resultado possível.

Autor



Mateus Espadoto

mespadoto@yahoo.com

Há 14 anos trabalhando com TI, já trabalhou em áreas tão diversas como desenvolvimento de software, infraestrutura e arquitetura corporativa. Atualmente trabalha como Administrador de Banco de Dados.



CURSOS ONLINE

A Revista .net Magazine oferece aos seus assinantes uma série de Cursos Online de alto padrão de qualidade.

.net
magazine



CONHEÇA OS CURSOS MAIS RECENTES:

- Curso Padrões de Projeto com C#
- Curso básico de ASP .NET
- Curso de Introdução ao .NET Framework
- Curso Básico de C#
- C# 5 e suas novidades
- ASP.NET MVC – Sistema de Vestibular

Para mais informações :

www.devmedia.com.br/curso/netmagazine

(21) 3382-5038



DEVMEDIA

Analizando o tráfego de redes com Wireshark

O que tem em sua rede?

O uso das redes de computadores em ambientes corporativos e educacionais se faz cada vez mais presente. Para que se mantenha um ambiente produtivo em funcionamento, torna-se necessário monitorar a rede, fazer o seu gerenciamento e administração. O Wireshark, capturador de pacotes antes conhecido como Ethereal, é provavelmente uma ferramenta indispensável para administradores de redes que queiram uma análise mais detalhada sobre o que trafega em suas redes. Com ele, todo o tráfego de entrada e saída é analisado e mostrado em uma lista com diversos recursos de navegação.

O Wireshark é uma ferramenta de rede passiva, ou seja, não transmite bytes para a rede, nem é o destinatário de bytes que outros computadores enviaram. Durante sua instalação, disponível tanto para sistemas operacionais baseados em Unix quanto em ambientes Windows, é possível selecionar componentes adicionais, a começar pela interface gráfica, como por exemplo:

- TShark, um analisador de protocolo de rede baseado em modo texto;
- Plugins e extensões que ainda estão em caráter experimental;
- Ferramentas de linha de comando;
- Guia do usuário.

De maneira geral, todo analisador de pacotes é composto por duas partes: o módulo de captura de pacotes (*Packet Capture Library*) e o analisador de protocolos (*Protocol Analyzer*). No módulo de captura, é feita a cópia de todos os pacotes que atravessam a placa de rede pré-especificada para a memória do computador, enquanto o analisador de protocolos interpreta os cabeçalhos e conteúdos dos pacotes nos vários níveis da arquitetura de camadas das redes de computadores, como a arquitetura do modelo ISO/OSI. Como a maioria dos programas de captura de dados, o Wireshark utiliza a biblioteca Libpcap (veja o **BOX 1**), que oferece uma interface independente de sistema operacional e compatível com diversas tecnologias para capturar pacotes.

Resumo DevMan

Porque esse artigo é útil:

Este artigo descreve conceitos fundamentais sobre o Wireshark, anteriormente conhecido como Ethereal, um aplicativo GPL analisador de tráfego de redes.

Um analisador de tráfego captura as informações que estão sendo transmitidas em uma determinada rede e apresenta de forma detalhada cada pacote de dados capturado, organizando-os por tipo de protocolo. O Wireshark pode ser útil para aqueles que queiram monitorar o que está acontecendo em sua própria rede, examinar problemas de segurança, resolver problemas de implementação de protocolos ou mesmo por pessoas que estejam interessadas em aprender sobre o funcionamento de algum protocolo de rede específico.

BOX 1. Libpcap

Muitas aplicações de apoio ao gerenciamento e segurança de redes fazem uso especificamente da captura passiva de tráfego, utilizando uma biblioteca de software chamada Libpcap, que é open source, portátil e provê funcionalidades para captura de tráfego de dados e protocolos de redes.

Por padrão, o Wireshark utiliza interface gráfica, mas existe uma opção em modo texto, chamada TShark. O TShark é uma ferramenta compatível com diversos sistemas e com os filtros realizados na versão gráfica do Wireshark, que permite implementar rotinas automatizadas usando linguagens de alto nível como Python e Perl. No Wireshark, a captura dos dados é feita em tempo real a partir de qualquer uma das interfaces de rede da máquina responsável pela coleta, como pode ser visto na **Figura 1**.

Na janela de interfaces, ao clicar no botão *Options* e selecionar a placa que será utilizada para fazer a captura, a opção *Capture all in promiscuous mode* deverá ser escolhida para que o programa capture todas as informações em modo promíscuo, ou seja, na máquina onde o Wireshark estiver instalado, a placa de rede irá capturar todos os pacotes da rede atual, mesmo que esta informação não tenha como destino esta máquina. Um ponto que deve ser observado é que nem todas as interfaces de rede suportam trabalhar em modo promíscuo, devido a limitações do próprio hardware.

Ao iniciar a ferramenta, de maneira geral, podem-se verificar no Wireshark cinco partes principais, como mostra a **Figura 2** e descritos a seguir:

1. Menu de comandos: são os menus localizados no topo da janela, como pode ser visto na **Figura 3**:

- **File:** expõe ao usuário ações básicas para abrir um arquivo de captura, salvar, exportar para os formatos suportados pelo Wireshark, entre outras funções;
- **Edit:** permite a marcação de pacotes, avançar a captura, busca e alterações na configuração do Wireshark;

- **View:** apresenta painéis e menus que podem ser visualizados e ocultos, configuração do tempo relativo, adição de colunas no painel de pacotes capturados e definir as cores por protocolos capturados;

- **Analyze:** apresenta para o usuário os filtros e protocolos que podem ser utilizados tanto na captura quanto na visualização das informações;

- **Capture:** lista as interfaces disponíveis para captura, opções de captura, iniciar, parar, reiniciar e filtros de captura;

- **Statistics:** permite que o usuário visualize estatísticas por protocolo, pacotes capturados, médias, filtro por tamanho do pacote, gráficos, contadores de pacotes, gráficos de fluxo, etc.;

- **Telephony:** é um menu dedicado aos protocolos utilizados pela tecnologia VoIP que mostra estatísticas de chamadas e parâmetros para medir a qualidade das chamadas;

- **Tools:** apresenta ferramentas para criar regras de filtro de pacotes;

- **Internals:** mostra ao usuário os protocolos e todos os filtros suportados pelo Wireshark;

- **Help:** é onde o usuário pode verificar a ajuda online, exemplos de capturas e versão da ferramenta.

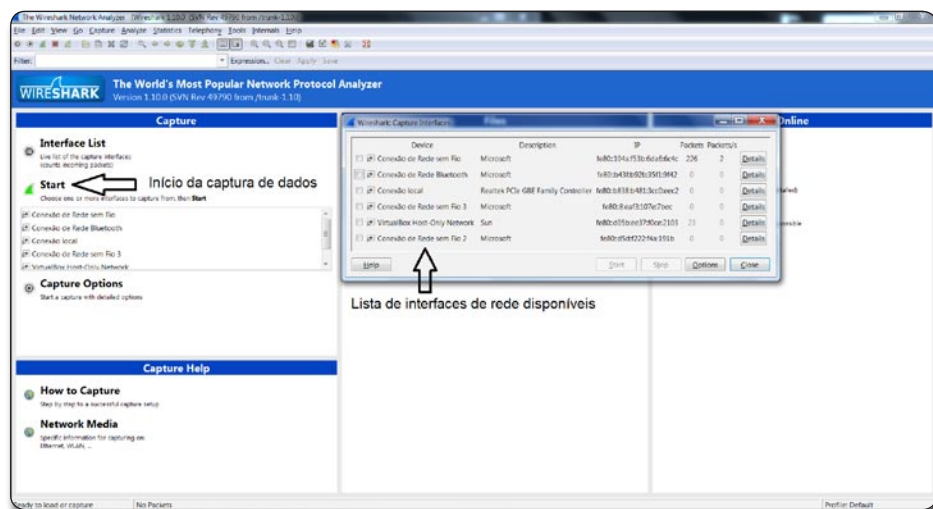


Figura 1. Lista de interfaces e início da captura de tráfego no Wireshark

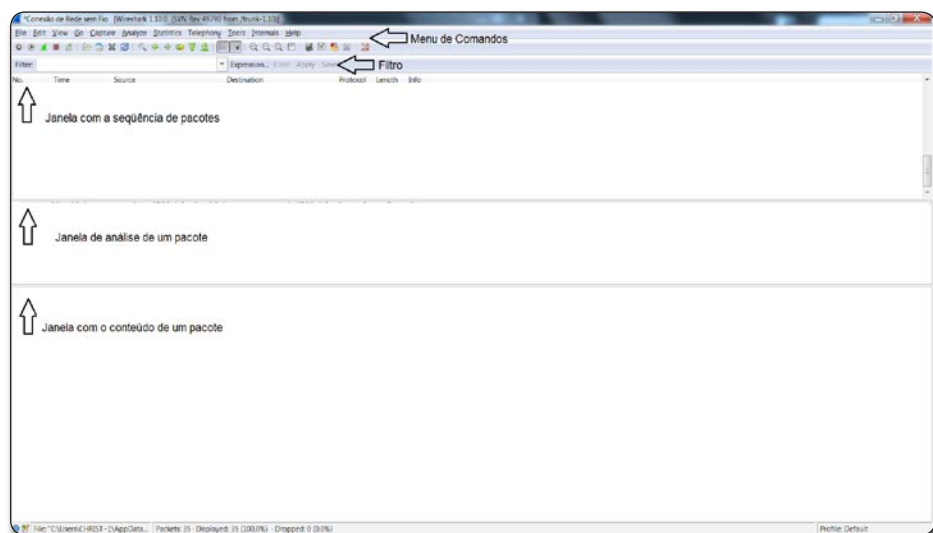


Figura 2. Visão Geral do Wireshark



Figura 3. Menu de Comandos disponíveis no Wireshark

2. Filtro: campo onde é possível inserir informações para especificar quais pacotes devem ser mostrados na janela com a sequência de pacotes;

3. Janela com a sequência de pacotes: apresenta uma linha para cada um dos pacotes capturados, com seu número (introduzido pelo Wireshark), instante em que foi capturado (em relação ao primeiro pacote), sua origem e destino, tipo de protocolo e alguma informação sobre o conteúdo do pacote;

4. Janela de análise de um pacote: apresenta detalhes sobre o pacote selecionado na janela com a sequência de pacotes, indicando os vários encapsulamentos a que foi sujeito, isto é, os cabeçalhos introduzidos pelas várias camadas. Detalhado nas colunas tem-se o número do pacote, tempo relativo, origem e destino do pacote, protocolo do pacote, tamanho e informações gerais do pacote. Estas colunas podem ser personalizadas;

5. Janela com o conteúdo de um pacote: contém o conteúdo do pacote selecionado na janela com a sequência de pacotes, em formato hexadecimal e ASCII.

Por padrão, o Wireshark só apresenta opções para análise completa dos pacotes depois de finalizada a captura, mas é possível acompanhar o fluxo de pacotes “ao vivo”, e parar a captura assim que o pacote de interesse for capturado.

O Wireshark pode exportar e importar dados em vários formatos, sendo um dos mais importantes o da Libpcap, que pode ser usado por várias ferramentas, o que facilita a leitura dos dados e posterior análise através de outros aplicativos compatíveis com esta biblioteca. Também é possível exportar os dados coletados para formatos como CSV, texto puro e PostScript, como pode ser visto na **Figura 4**.

Análise das informações capturadas

O Wireshark permite que se faça uma série de análises com base nos dados coletados. O item *Analyze*, selecionado na **Figura 5**, mostra algumas opções de criação e edição de filtros que podem, por exemplo, exibir detalhes de um determinado pacote. Este item permite também habilitar/desabilitar protocolos, criar regras de comando ACL (*Access Control List* – ver **BOX 2**) para vários firewalls diferentes e exibe informações de um determinado protocolo (TCP, UDP ou SSL) que são capturadas na mesma conexão que o pacote selecionado.

BOX 2. ACL

Access Control List ou Lista de Controle de Acesso é uma lista que define quem tem permissão de acesso a certos serviços da rede, determinando tipos de acesso para cada usuário ou grupo. Roteadores utilizam ACLs para filtragem de pacotes, seja ele de entrada ou de saída, TCP/UDP, entre outros protocolos.

Com o Wireshark é possível, por exemplo, verificar os sites que o usuário está visitando, assim como obter informações sobre qual sistema operacional e navegador o usuário está utilizando, conforme verificado na **Figura 6**. Como pode ser visto na linha do protocolo selecionado, no caso o protocolo HTTP, no campo que exibe o conteúdo deste pacote podemos observar que a máquina que fez este acesso é uma máquina com Mac OS X e o navegador Mozilla Firefox.

Estatísticas dos protocolos coletados

O Wireshark provê uma grande variedade de estatísticas sobre a rede, disponibilizando informações gerais a respeito da

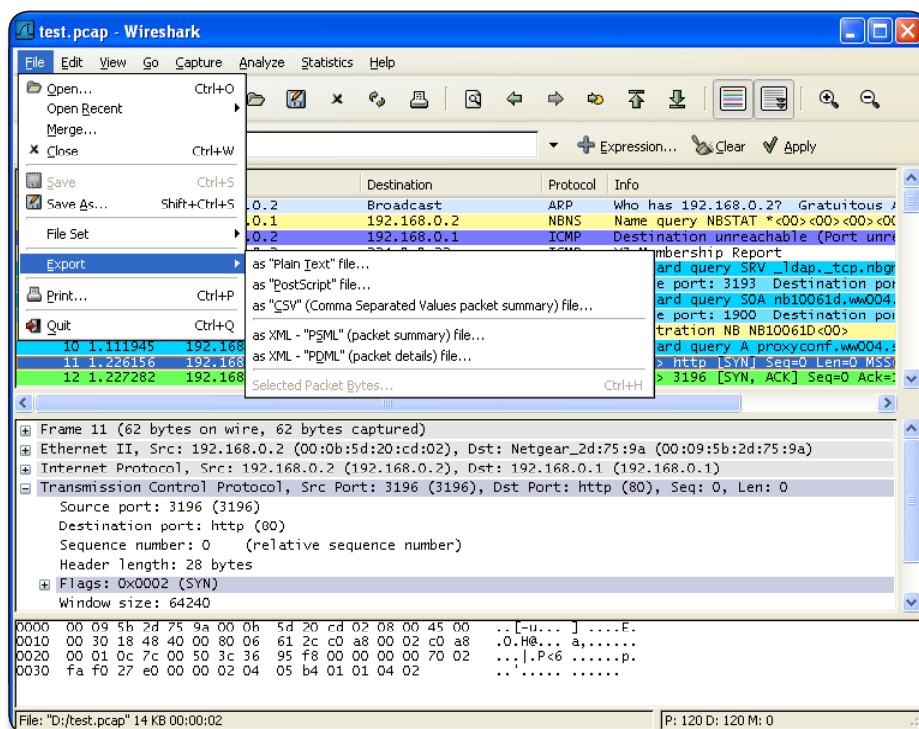


Figura 4. Exportando informações capturadas no Wireshark

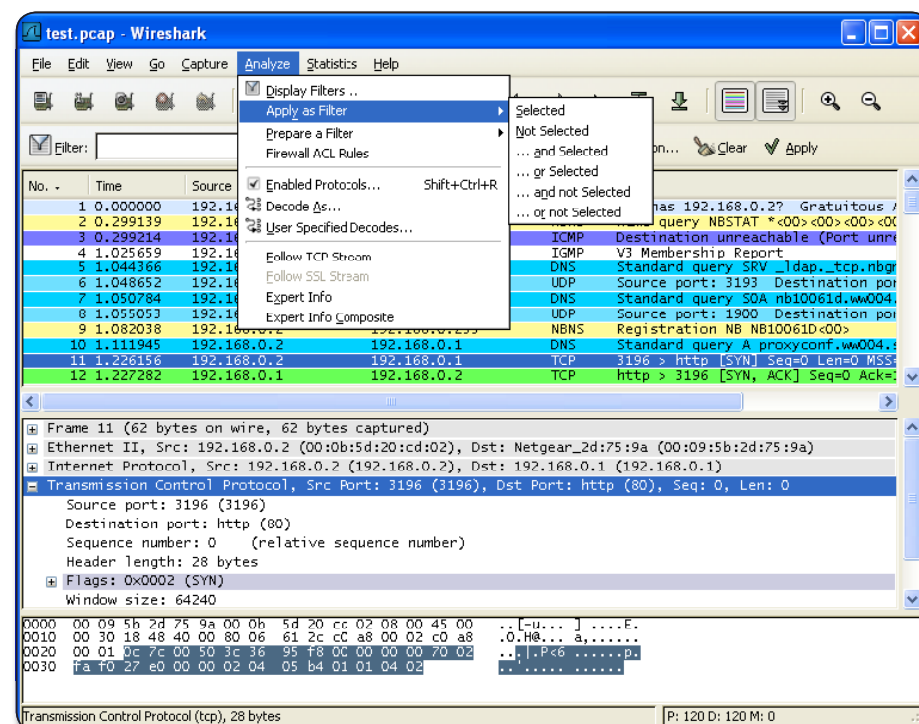


Figura 5. Tipos de Análises no Wireshark

captura dos dados da rede em análise, o número de pacotes capturados, estatísticas sobre protocolos específicos, hierarquia de protocolos coletados, tráfego de/para IPs

específicos na rede, incluindo outras máquinas ou servidores, e apresenta gráficos para melhor visualização das informações sobre a rede.

Na opção *Hierarchy Statistics*, um subitem do menu *Statistics*, é possível verificar todos os protocolos capturados pelo Wireshark, com porcentagem individual para cada tipo encontrado, como demonstrado no exemplo de análise mostrado na **Figura 7**. Este recurso apresenta diversas

estatísticas que estão disponíveis sobre cada captura, a saber:

- **Protocol:** informa o nome do protocolo;
- **% Packets:** apresenta em forma de porcentagem a quantidade de pacotes do protocolo que foram capturados no total;
- **Packets:** determina o número absoluto

de pacotes deste protocolo que foram encontrados na captura;

- **Bytes:** determina o tamanho em bytes dos pacotes deste protocolo;
- **MBit/s:** apresenta a quantidade de bits capturados por segundo de um determinado protocolo.

O termo Endpoint no Wireshark é uma marcação que representa um ponto final entre dois endereços, levando em consideração um determinado protocolo ou interface de rede, com a finalidade de verificar o que foi enviado/recebido entre dois pontos definidos. Já em outras ferramentas de rede, os endpoints são chamados de host list ou lista de hosts. Para fins de análise, as estatísticas do Wireshark irão considerar os seguintes endpoints:

- **Ethernet:** corresponde ao endereço MAC;
- **FDDI:** corresponde ao endereço MAC FDDI;
- **IPv4:** corresponde ao endereço IP;
- **TCP:** combinação entre endereço IP e porta TCP usada. Vale ressaltar que diferentes portas TCP no mesmo endereço IP indicam endpoints TCP diferentes;
- **Token Ring:** endereço MAC do Token Ring;
- **UDP:** combinação do endereço IP e da porta UDP.

Quando se tem uma comunicação na rede, o Wireshark interpreta como sendo um tráfego entre dois endpoints específicos, ou seja, duas interfaces de rede ou endereços IPs específicos que estão se comunicando e trocando informações entre si. A janela de conversações pode ser acessada no menu *Statistics > Conversation*, onde se tem o endereço A e o endereço B dos dois endpoints que estão sendo analisados.

A **Figura 8** mostra uma conversação Ethernet, a **Figura 9** uma conversação IPv4 e a **Figura 10** uma conversação UDP. Nessas conversações é possível verificar qual a quantidade de pacotes e bytes que foram recebidos no total, qual a quantidade de pacotes e bytes que foram recebidos no total do endereço A para o endereço B e vice-versa, qual a duração da comunicação entre os endpoints e a quantidade de

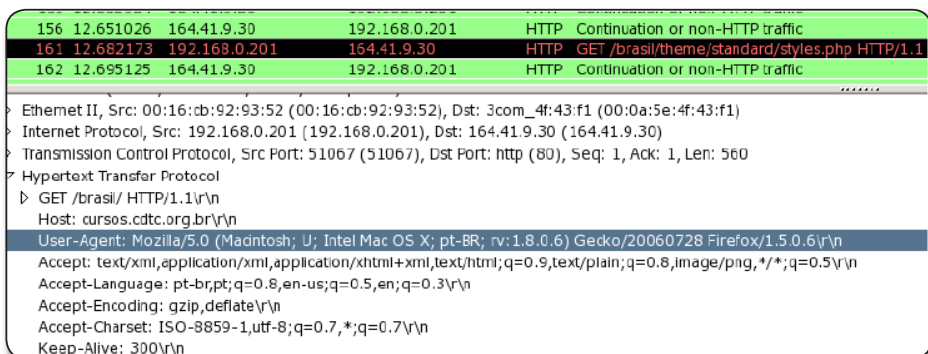


Figura 6. Análise de pacote no Wireshark: sistema operacional e navegador do usuário capturado

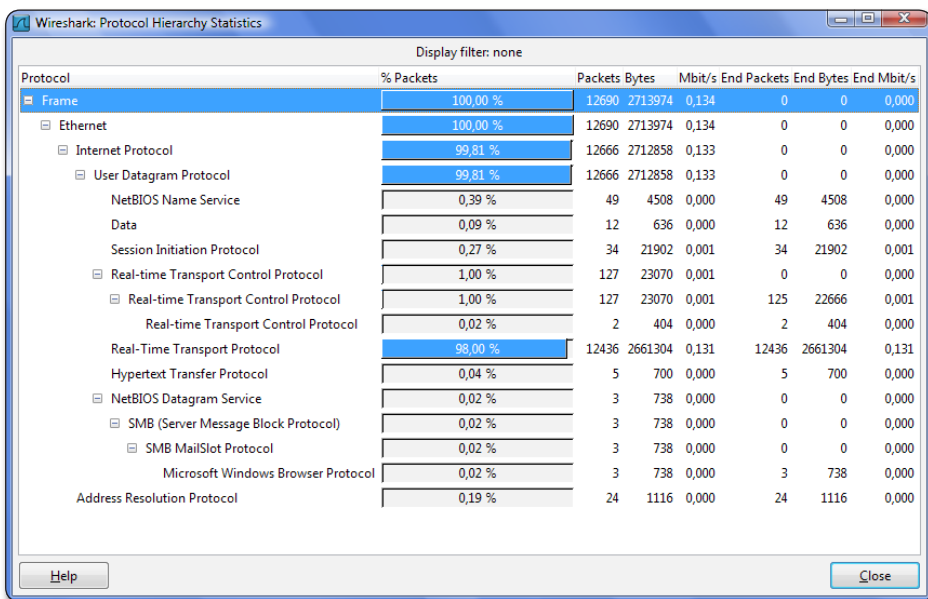


Figura 7. Exemplo de protocolos capturados no Wireshark

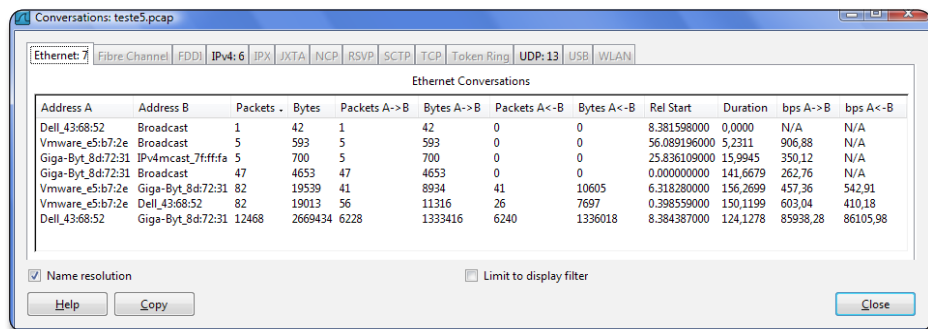


Figura 8. Exemplo de conversação Ethernet no Wireshark

tráfego em bits por segundo do endereço A para o endereço B e vice-versa.

Com isso, é possível verificar se o endereço A ou o endereço B estão realmente conectados e transmitindo dados entre si, se houve ou não perda de dados na comunicação, visualizar dados sobre as interfaces de rede dos endereços A e B, entre outras informações.

Utilizando filtros

Dependendo da rede analisada, a quantidade de informações pode ser incrivelmente grande. Pensando nesta possível dificuldade, um dos recursos mais poderosos do Wireshark é justamente a possibilidade de se criar filtros para limitar o número de pacotes visíveis, para que a quantidade de informações não úteis não seja “ensurdecadora”. Para limitar as informações e ser mais específico no que será analisado, é possível construir expressões para realizar essa filtragem, como pode ser visto na **Figura 11**. O próprio site do Wireshark também possui uma lista com vários filtros disponíveis para análise de alguns protocolos. Se algum novo protocolo for criado pelo IETF, existe a possibilidade de inserção de novos filtros.

Ao abrir a janela de expressões para os filtros é exibida uma lista com os nomes dos campos organizada por protocolos e uma caixa para selecionar as relações que se deseja analisar. Clicando no “+” do lado do protocolo, aparece uma lista com os tipos de filtros para este protocolo, e na caixa de relação existe uma lista de operadores disponíveis (“==”, “!=”, “>”, etc.). A relação *is present*, por exemplo, é uma relação unária que é verdadeira quando o campo selecionado está presente no pacote. Todas as outras relações são binárias e precisam de dados adicionais. Quando uma delas é selecionada, torna-se indispensável digitar um parâmetro que deverá ser utilizado para a comparação. Por exemplo, alguns protocolos já têm parâmetros pré-definidos por serem os mais utilizados, sendo necessário escolher algum destes valores para criar o filtro desejado, como pode ser visto nos exemplos a seguir.

Address A	Address B	Packets	Bytes	Packets A->B	Bytes A->B	Packets B->A	Bytes B->A	Rel Start	Duration	bps A->B	bps B->A
192.168.1.223	192.168.1.255	5	593	5	593	0	0	56.089196000	5,2311	906,88	N/A
192.168.1.120	239.255.255.250	5	700	5	700	0	0	25.836109000	15,9945	350,12	N/A
192.168.1.120	192.168.1.255	47	4653	47	4653	0	0	0.000000000	141,6679	262,76	N/A
192.168.1.120	192.168.1.223	70	18509	20	7445	50	11064	0.398559000	150,1199	396,75	589,61
192.168.1.120	192.168.1.223	76	19233	38	10425	38	8808	6.318280000	156,2699	533,69	450,91
192.168.1.100	192.168.1.120	12463	2669170	6226	1333332	6237	1335838	8.384424000	124,1277	85932,89	86094,41

Figura 9. Exemplo de conversação IPv4 no Wireshark

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A->B	Bytes A->B	Packets B->A	Bytes B->A	Rel Start	Duration	bps A->B	bps B->A
192.168.1.223	netbios-dgm	192.168.1.255	netbios-dgm 1	2	225	1	225	0	0	61.201232000	0,0000	N/A	N/A
192.168.1.120	netbios-dgm	192.168.1.255	netbios-dgm 2	2	513	2	513	0	0	83.460432000	53,6182	76,54	N/A
192.168.1.223	netbios-ns	192.168.1.255	netbios-ns 4	4	368	4	368	0	0	56.089196000	2,2301	911,44	N/A
192.168.1.120	17141	239.255.255.250	ssdp	5	700	5	700	0	0	25.836109000	15,9945	350,12	N/A
192.168.1.100	43146	192.168.1.223	15912	22	4708	5	1070	17	3638	8.265278000	0,3096	27647,33	94000,92
192.168.1.120	30275	192.168.1.223	sip	23	10995	14	5341	9	5654	6.318280000	156,2699	273,42	280,45
192.168.1.100	36434	192.168.1.223	sip	23	11543	14	6201	9	5342	0.398559000	150,1199	330,46	264,68
192.168.1.100	43147	192.168.1.223	15913	25	2256	1	174	24	2084	8.255682000	120,0164	N/A	138,91
192.168.1.120	51825	192.168.1.223	17165	26	2460	2	376	24	2084	8.233369000	124,3901	24,18	134,03
192.168.1.120	51824	192.168.1.223	17164	27	5778	22	4708	5	1070	8.264873000	124,3467	302,90	68,84
192.168.1.120	netbios-ns	192.168.1.255	netbios-ns 45	45	4140	45	4140	0	0	0.000000000	141,6679	233,79	N/A
192.168.1.120	51825	192.168.1.100	43147	76	18520	40	9680	36	8672	11.304260000	121,1524	639,19	572,63
192.168.1.120	51824	192.168.1.100	43146	12387	2650818	6197	1326158	6190	1324660	8.384424000	124,1277	85470,73	85373,99

Figura 10. Exemplo de conversação UDP no Wireshark

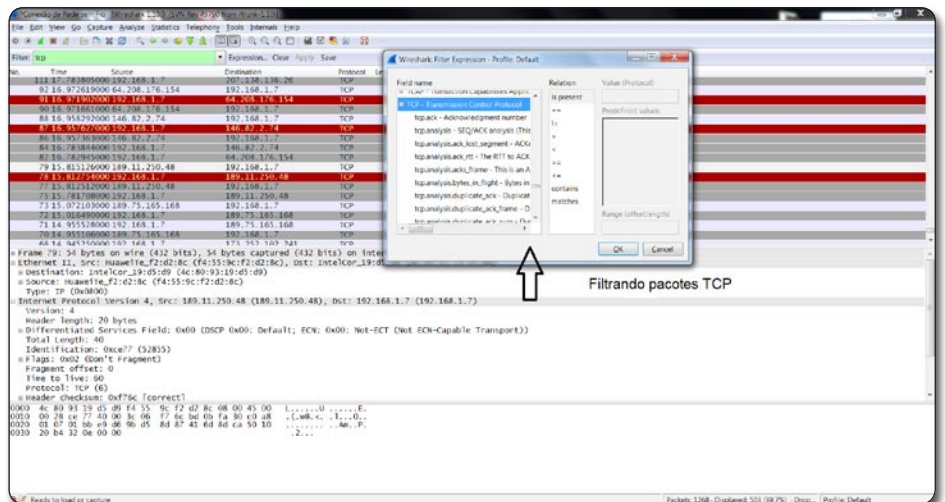


Figura 11. Filtragem de pacotes TCP no Wireshark

Os operadores lógicos que podem ser empregados junto aos filtros são apresentados na **Tabela 1**.

Como exemplo, pode-se digitar no campo *Filter* do Wireshark os seguintes comandos para filtrar o tráfego da web:

`tcp.dstport == 80 or tcp.dstport == 443`

ou

`tcp.port == 80 and ip.src == 192.168.1.1`

ou

`http.host eq "www.mulheresnatecnologia.org"`

No primeiro exemplo, é possível filtrar apenas os pacotes trafegados pela porta 80 (utilizada pelo protocolo HTTP) ou 443 (utilizada pelo protocolo HTTPS) do

protocolo TCP. O segundo exemplo permite filtrar o tráfego da porta 80 do protocolo TCP vinda do endereço 192.168.1.1. Já o terceiro exemplo de filtro permite selecionar todas as informações capturadas do host “www.mulheresnatecnologia.org”.

Outro filtro pode selecionar componentes específicos de uma conexão HTTP, como por exemplo, um código de resposta (mostrar apenas erros 404):

```
http.response.code == 404
```

As capturas feitas pelo TShark também utilizam a mesma sintaxe da Libpcap, conforme demonstra o comando:

```
TShark -f “filtro”
```

Nele, filtro é o que se deseja selecionar no tráfego capturado. Por exemplo, para apresentar qualquer pacote UDP que use a porta 53, o comando seria:

```
TShark -f “udp port 53”
```

Já para filtrar respostas HTTP 404 (página não encontrada), como mostrado no Wireshark, a sintaxe para o TShark ficaria:

```
TShark -R “http.response.code == 404”
```

O Wireshark também permite filtros específicos para analisar chamadas VoIP. Em *Telephony – Voip Calls*, para preparar um filtro para uma determinada chamada, basta selecionar a chamada desejada e pressio-

nar o botão *Prepare Filter*. Isto irá criar um filtro na janela principal do Wireshark para selecionar os pacotes relacionados a essa chamada em análise, como mostra a **Figura 12**. A partir desses dados é possível verificar características do ambiente VoIP e monitorar se a comunicação está dentro dos parâmetros de inteligibilidade e disponibilidade para os usuários.

Após aplicar o filtro para listar as chamadas VoIP encontradas na captura, tem-se as seguintes informações por chamada:

- **Start Time:** Hora de início da chamada;
- **Stop Time:** Hora de término da chamada;
- **Initial Speaker:** O IP de origem do pacote que iniciou a chamada;
- **From:** Para chamadas H323 e ISUP, este é o número de quem originou a chamada. Para chamadas SIP, é o campo “De” do convite;
- **To:** Para chamadas H323 e ISUP, este é o número chamado. Para chamadas SIP, é o campo “Para” do convite;
- **Protocol:** Define um dos protocolos listados acima;
- **Packets:** Apresenta o número de pacotes envolvidos na chamada;
- **State:** Estado da chamada atual. Os valores possíveis são CALL SETUP (chamada em estado de configuração); RINGING (chamada tocando); IN CALL (chamada em curso); CANCELLED (chamada foi liberada antes de conectar com o usuário de destino); COMPLETED (chamada foi efetuada e, em seguida, liberada); REJECTED (chamada foi liberada antes de conectar ao destino); e UNKNOWN (chamada em estado desconhecido).

Operadores	Símbolos	Significado
eq	==	Igual
ne	!=	Diferente
gt	>	Maior
lt	<	Menor
ge	>=	Maior ou Igual
le	<=	Menor ou Igual
and	&&	Lógico AND
or		Lógico OR
xor	^^	Lógico XOR
not	!	Lógico NOT

Tabela 1. Operadores e valores lógicos utilizados para a criação de filtros no Wireshark

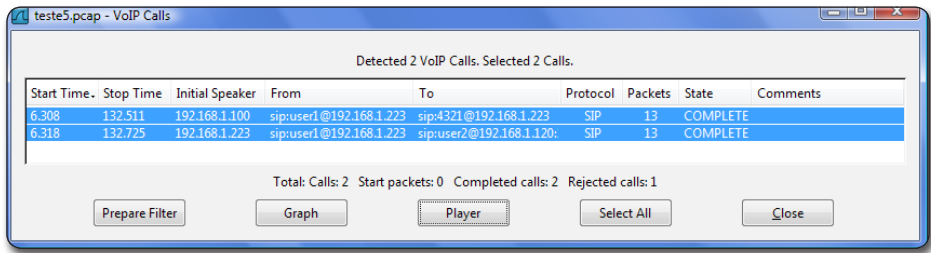


Figura 12. Filtragem de chamadas VoIP no Wireshark

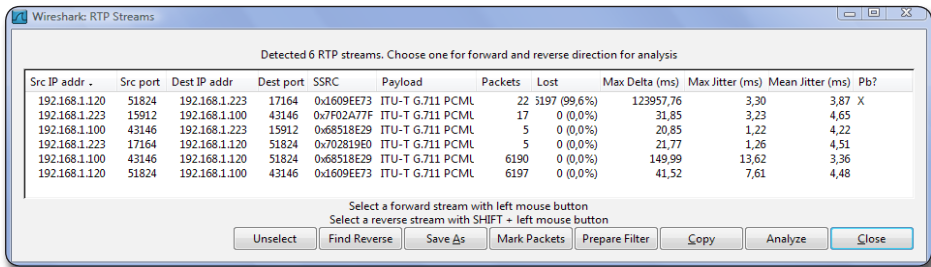


Figura 13. Informações sobre chamadas VoIP no Wireshark

Na **Figura 13** é possível visualizar os dados de uma conversação VoIP que utiliza o protocolo RTP, como por exemplo, os tipos de codecs da conversação, a perda de pacotes e o atraso. Com esses dados coletados temos informações úteis para prover ajustes para a melhoria da comunicação. Visando auxiliar esta configuração, na literatura existem parâmetros de referência para verificar a qualidade do serviço de comunicação VoIP (veja a **Tabela 2**).

Atraso da Voz Fim a Fim (ms)	Varição do Atraso (ms)	Média das Perdas de Pacotes (%)	Tolerância
Até 150	0	1 a 2	Aceitável, com boa interatividade
150 – 400	75 a 125	3 a 15	Aceitável, mas o usuário já percebe alguma perda de interatividade
Acima de 400	225	25	Inaceitável. Perda da interatividade

Tabela 2. Parâmetros utilizados como referência para medir a qualidade de uma comunicação VoIP

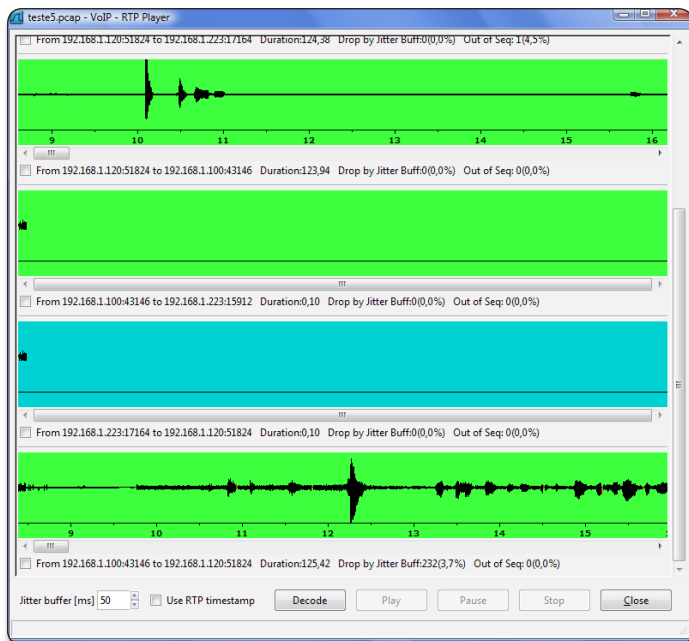


Figura 14. Player para remontagem de chamadas VoIP no Wireshark

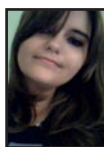
O Wireshark apresenta ainda um player para dados do protocolo RTP que permite remontar a conversação realizada em servidores VoIP que não utilizam nenhum mecanismo de criptografia, como apresentado na **Figura 14**. A partir deste player é possível ouvir a conversa original na íntegra.

Conclusão

O Wireshark é uma ferramenta livre que permite a análise de protocolos, podendo auxiliar administradores de rede na resolução

de problemas. Ele realiza a captura e análise de informações trafegadas em diferentes tipos de meios de transmissão, sejam eles cabeados ou não, com a vantagem de compatibilidade entre os mais diversos sistemas operacionais. Além disso, por meio do Wireshark, torna-se possível conhecer diversas características e o funcionamento de vários tipos de protocolos. O conhecimento aprofundado dos protocolos proporciona a criação de novas ferramentas, uma melhor utilização dos recursos de cada protocolo e maior compatibilidade entre tecnologias.

Autor



Christiane Borges Santos

christiane.borges@ifg.edu.br

Atua como docente na área de Redes de Computadores há mais de cinco anos. É tecnóloga em Redes de Comunicação, pelo CEFET Goiás, mestre em Engenharia Elétrica e da Computação, pela UFG. Atualmente cursa doutorado em Ciências da Computação, na UnB. É professora adjunta no IFG e membro do comitê gestor do grupo /MNT (Mulheres na Tecnologia).

Links:

Wireshark - What's on your network? (O que tem em sua rede?)

<http://www.wireshark.org>

Referências de Filtros no Wireshark

<http://www.wireshark.org/docs/dfref/>

IETF – Internet Engineering Task Force

<http://www.ietf.org>

Pentest: avaliação do nível de segurança de uma rede

Um estudo para evitar erros na contratação e deixar de lado a falsa sensação de segurança

Nos últimos anos as empresas brasileiras adotaram massivamente a tecnologia buscando atender as necessidades de conectividade e mobilidade cada vez mais exigidas para a evolução de seus negócios. Para alguns setores do mercado, manter a disponibilidade de informações e serviços aos seus funcionários, fornecedores e clientes, há muito tempo deixou de ser um diferencial competitivo para se tornar uma necessidade básica e obrigatória, fundamental para sua existência. Mesmo que algumas atividades possam parecer não depender diretamente de apoio da área de TI, o negócio como um todo com certeza depende. Um exemplo muito claro é o que vemos em muitas metalúrgicas, com suas inúmeras máquinas ejetando milhares de peças sem parar. Empresas assim mantêm equipes de mecânicos de manutenção sempre apostos, pois a produção não pode parar caso alguma máquina emperre uma de suas engrenagens, pois na visão de seus diretores, os maquinários são os ativos responsáveis pelo lucro da empresa e, portanto devem estar sempre disponíveis. Pois bem, do que adiantará ter atingido a meta da linha de produção se o sistema que emite as notas fiscais dos produtos não estiver disponível quando as peças precisarem ser despachadas nos caminhões?

Garantir, além da disponibilidade, fatores como integridade e confidencialidade das informações estratégicas da organização, são exigências cada vez mais solicitadas aos departamentos de TI, e como os ambientes onde são processadas, transmitidas e armazenadas estas informações sofrem constantes alterações físicas e lógicas, é factível que os controles de segurança aplicados até o momento das alterações possam não estar mais protegendo adequadamente estes ativos.

Resumo DevMan

De que se trata o artigo:

Este artigo abordará pontos importantes para auxiliar na compreensão de um termo que vem sendo amplamente usado no mercado, mas algumas vezes de forma equivocada ou incompleta. Pentest é o acrônimo de Penetration Test (Teste de Penetração) e como poderemos ver, trata-se de um procedimento que avalia o nível de segurança de uma rede, sistema ou aplicação, externa ou internamente, com a ciência ou não da equipe responsável pela proteção do ativo, fazendo uso de ferramentas e até de práticas de Engenharia Social.

Compreender minimamente os conceitos que envolvem um Pentest ajudará os leitores a evitar gastos desnecessários com a aquisição de soluções e serviços que darão uma falsa sensação de segurança. Muitas empresas atuam de forma equivocada na hora de vender um serviço de Pentest aos seus clientes, automatizando processos e gerando relatórios genéricos sem uma análise mais profunda dos resultados da varredura, indicando onde e como as determinadas vulnerabilidades poderão afetar o negócio do cliente.

A questão é: como descobrir isso antes da ocorrência de um incidente?

No cenário ideal, descobriríamos as falhas por meio de testes sistemáticos, simulando situações de ataques ou falhas operacionais que poderiam levar a uma indisponibilidade em determinados sistemas ou a corrupção de dados críticos, como uma base de dados com o cadastro de todos os clientes. Estes testes ajudariam a detectar vulnerabilidades até então desconhecidas pelos administradores da rede, assim como permitir aos gestores do negócio ter uma estimativa de possíveis prejuízos se tal incidente ocorresse. Também daria ao time de resposta a incidentes uma visão

de quais ações necessitariam ser tomadas na concretização deste fato e mais do que isso, poderia ajudar a equipe de TI a adotar controles que mitigassem estes riscos. Todo este esforço no final das contas sairia mais barato do que uma única interrupção que pudesse afetar o negócio. Muitas vezes o prejuízo de uma interrupção não atinge somente a parte financeira da empresa, mas também sua imagem e reputação de forma negativa, provocando desgastes nas equipes envolvidas no problema.

Mas o que vemos na prática na maioria das empresas é o reflexo do padrão cultural do brasileiro, que prefere tomar ações reativas a preventivas. Prefere correr o risco de passar por um incidente do que criar meios para tentar evitá-lo, ou seja, somente quando a rede é atacada, quando um sistema entra em falha e quando dados são perdidos é que se pensa em adotar os devidos controles de segurança. No entendimento da alta direção, isso nunca irá acontecer, pois ela gastou alguns milhares de reais em “firewalls” e “antivírus” e, portanto a rede está mais do que protegida. Porém, o que estes não compreendem é que para se ter um ambiente seguro não adianta colocar um firewall de última geração “padrão NASA” ou “padrão FIFA”, como vimos recentemente no país. Também não adianta comprar aquele antivírus com mais propagandas no mercado se a equipe que for administrá-lo não estiver plenamente capacitada, com disponibilidade para gerenciar as soluções e principalmente com seus controles de segurança alinhados com os requisitos e expectativas do negócio.

Como consequência desta má gestão, no primeiro grave incidente que paralise as operações da empresa, a alta direção baterá na porta do departamento de TI questionando “todo aquele investimento” em firewalls e antivírus, deixando a impressão de que não vale a pena “gastar” com TI. Analisando a situação, até entendemos porque a TI acaba sendo vista pelos demais departamentos como um ralo, escoando dinheiro da organização, pois como justificar que determinada falha ocorreu após a implementação de controles de segurança caríssimos, como firewalls de última geração ou soluções de detecção de intrusão? Como a TI pode ser pega de surpresa numa situação como esta? A resposta é a falta de prevenção, algo enraizado na cultura dos profissionais brasileiros, que reflete diretamente na gestão destas empresas. Infelizmente, amigos leitores, este ainda é o cenário dominante nas organizações públicas e privadas de nosso país.

Porém há uma luz lá no fim do túnel que começa a trazer um pouco de esperança de que esta cultura possa ser modificada, mesmo que seja na base da obrigação, na necessidade de cumprir regras. A antiga Norma ISO 17799, hoje oficialmente conhecida como ABNT NBR ISO/IEC 27002 – Tecnologia da Informação – Técnicas de Segurança – Código de Prática Para a Gestão da Segurança da Informação, lançada em 2005 (e atualizada em 2007), já citava a necessidade de atender a requisitos de auditorias de sistemas com o objetivo de minimizar o risco de interrupção dos processos do negócio. Esta norma ainda não é obrigatória, como uma ISO 9001, porém a cada dia empresas com certo grau de maturidade em seus processos começam a adotar determinados controles e também começam a exigir de seus parceiros os mesmos cuidados.

Empresas de diferentes setores do mercado financeiro como bancos e operadoras de cartões de crédito, cansadas de acumularem prejuízos com fraudes eletrônicas, iniciaram em Setembro de 2006 a criação de um Conselho para debater e elaborar normas contendo as melhores práticas para o manuseio e armazenagem de dados de cartões de crédito, a *Payment Card Industry (PCI) – Data Security Standard (DSS)*, mais conhecida como PCI-DSS. Isso está fazendo com que empresas que possuam operações de comércio eletrônico ou que prestam serviços a instituições financeiras comecem a adotar os requisitos da PCI-DSS, com destaque para o *Requisito 11: Testar regularmente os sistemas e processos de segurança*.

Neste requisito em específico, a Norma PCI-DSS exige que as empresas realizem vários tipos de testes periodicamente na rede interna e externa, incluindo avaliações nos pontos de acesso wireless, firewalls e aplicações. Estas varreduras e testes de penetração em alguns casos devem ser realizados trimestralmente ou após alguma modificação no ambiente de rede.

Desta forma, a busca por serviços relacionados a revisões dos procedimentos de segurança e especificamente dos controles de segurança aumentará de forma significativa, incluindo principalmente auditorias técnicas como os Testes de Penetração. Porém, no mercado a divulgação deste tipo de serviço já acontece há algum tempo e a enxurrada de termos técnicos e siglas podem confundir até mesmo os gestores de TI mais experientes. Em meio a esta confusão, encontramos inúmeras empresas e especialistas que oferecem serviços a custos estratosféricos, simplesmente para rodar uma aplicação automatizada e gerar um relatório padrão, sem qualquer análise mais aprofundada.

Para garantir a correta contratação deste tipo de serviço, adequado à necessidade de cada empresa, com a qualidade reconhecida no mercado e principalmente dentro de uma margem de preço aceitável ao tipo de serviço contratado, apresentaremos aos leitores alguns pontos básicos que todos os gestores que necessitam deste serviço deveriam entender antes de adquirir um Teste de Penetração:

- O que é Pentest?
- Como é feito?
- Quando deve ser feito?
- Onde deve ser realizado?
- Por que tenho que fazer este teste?

O que é Pentest?

Teste de Intrusão, Teste de Invasão, Teste de Penetração ou simplesmente Pentest (acrônimo para Penetration Test) é um procedimento legal e autorizado que visa identificar por meio de uma série de testes em um ambiente de rede, em um sistema ou em uma aplicação, possíveis vulnerabilidades que possam ser exploradas a partir de uma conexão na rede interna ou externa, permitindo acesso não autorizado a informações, dados e dispositivos na rede.

É muito comum encontrar profissionais e empresas no mercado que distinguem o Pentest de uma Análise de Vulnerabilidades. Segundo estes, a Análise de Vulnerabilidades é um procedimento que

apenas faz uso de ferramentas automatizadas de escaneamento da rede para detectar vulnerabilidades pré-definidas em uma base de dados carregada por estas ferramentas, mas sem realizar a exploração destas falhas ou mesmo apontar recomendações para mitigação dos riscos de uma possível exploração. Já o Pentest, segundo eles, tem como objetivo realizar além do escaneamento, a exploração e a indicação da correção em relatórios detalhados. De acordo com o dicionário, o verbete *análise* consiste em examinar de forma detalhada cada parte de um todo, buscando compreender tudo aquilo que o caracteriza. Portanto, uma Análise de Vulnerabilidades não pode ficar restrita a uma simples varredura automática. Se fosse este o conceito, então o correto seria chamar de Varredura ou Escaneamento de Vulnerabilidades. Algumas metodologias incluem a Análise de Vulnerabilidades como uma parte integrante de um Teste de Penetração, isto é, uma etapa a ser cumprida.

O Pentest consiste em uma série de ações que faz uso de ferramentas, processos e pessoas para chegar a um conjunto de informações relevantes que possam auxiliar a organização na identificação e mitigação dos riscos.

Como é feito?

Não existe uma receita única, pronta para ser aplicada em qualquer ambiente de rede. Cada cenário deve ser avaliado e seu escopo definido em conjunto com o cliente, especificando os ativos a serem testados, o tipo de abordagem, o nível de aprofundamento dos testes, as definições de prazos de execução e a entrega do relatório. Os resultados deverão ser descritos de forma clara e objetiva, com a apresentação das recomendações corretivas e preventivas para eliminar ou reduzir os riscos de uma possível invasão na rede.



Figura 1. Principais metodologias aplicadas em Testes de Penetração

Porém, mesmo não havendo uma receita de bolo pronta, existem metodologias que podem e devem ser aplicadas para chegarmos aos resultados mais conclusivos, reduzindo a possibilidade de surgirem falsos positivos e erros de interpretação dos dados. Mesmo assim, ainda encontramos muitas empresas que adotam metodologias próprias, o que dificulta a validação dos resultados.

Na Figura 1 podemos ver a representação do logotipo das principais metodologias presentes no mercado.

A seguir, apresentaremos alguns pontos relevantes sobre cada uma das metodologias, para que o leitor possa compreender a inviabilidade de se trabalhar somente com uma única metodologia ou adotar metodologias próprias.

OSSTMM – Open Source Security Methodology Manual

A metodologia OSSTMM é desenvolvida pela ISECOM (Institute for Security and Open Methodologies), uma organização que realiza pesquisas na área de segurança de forma colaborativa, aberta e sem fins lucrativos. Trata-se de um manual de referência que pode ser aplicado a qualquer tipo de organização, independentemente do seu setor de atuação e tamanho. Abrange todas as áreas da segurança da informação, envolvendo a parte da segurança física, lógica e humana.

Outro ponto importante nesta metodologia é a realização do teste alinhada com a necessidade da organização, variando de acordo com o conhecimento prévio que o auditor possui em relação aos ativos que serão testados, assim como o nível de conhecimento que o alvo possui em relação ao teste a ser executado, conforme podemos verificar na Tabela 1.

De todos os tipos de teste apresentados na Tabela, o que mais se aproxima da realidade de uma tentativa de invasão realizada por um cracker é o tipo *Double Blind*, onde o atacante não tem informações do alvo e o alvo não sabe que será atacado.

OWASP – Open Web Application Security Project

A OWASP (ou Projeto Aberto de Segurança em Aplicações Web) é uma entidade sem fins lucrativos, composta por diversos profissionais voluntários das áreas de tecnologia e segurança da informação. Atualmente possui capítulos (filiais) espalhados em diversos países, entre eles o Brasil, com a finalidade de promover melhorias no desenvolvimento de sistemas, especialmente voltados ao ambiente Web. Além dos profissionais

TESTE	AUDITOR	ALVO/CLIENTE	CARATERÍSTICAS
Blind	Não conhece nada sobre o alvo.	Sabe do pentest e como será feito o ataque.	A equipe de TI pode se preparar antes.
Double Blind	Não conhece nada sobre o alvo.	Não sabe do pentest e nem como será atacado.	Próximo à realidade de invasão por crackers.
Gray Box	Conhecimento parcial do alvo.	Sabe do pentest e como será feito o ataque.	O pentest é realizado em ambiente monitorado.
Double Gray Box	Conhecimento parcial do alvo.	Sabe do pentest, mas não sabe como será atacado.	Simula ataques partindo de funcionários internos com privilégios de acesso.
Tandem	Conhecimento total sobre o alvo.	Sabe do pentest e como será feito o ataque.	Similar a uma auditoria, auxiliando na melhoria dos controles.
Reversal	Conhecimento total sobre o alvo.	Não sabe do pentest e nem como será atacado.	Testa a capacidade de resposta a incidentes.

Tabela 1. Tipos de Teste de Vulnerabilidade/Intrusão

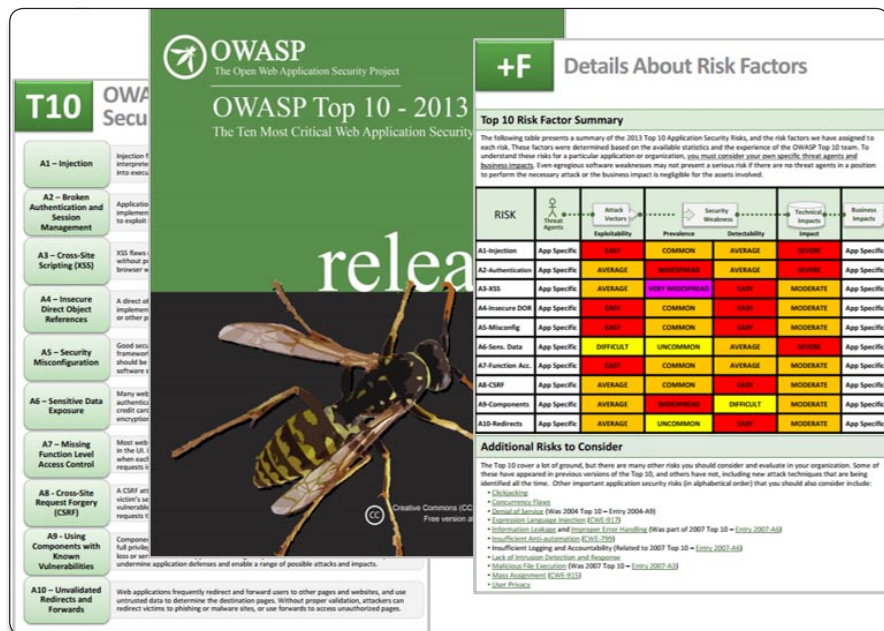
voluntários, grandes empresas de tecnologia ajudam a manter o projeto, como a Amazon, Fundação Mozilla, IBM, Oracle, entre outras. De todos os projetos desenvolvidos, um em especial tem ganhado espaço no mercado como uma referência tanto para desenvolvedores como para analistas de segurança: “OWASP TOP 10”. Trata-se de um estudo atualizado a cada três anos contendo informações sobre as principais vulnerabilidades em aplicações Web, com o objetivo de auxiliar as organizações na adoção de práticas seguras de desenvolvimento de software, apresentando recomendações para prevenir a exploração destas ameaças. Na Figura 2 podemos ver uma imagem do relatório de 2013.

NIST – National Institute Standards and Technology

O NIST (Instituto Nacional de Padrões e Tecnologia) é um órgão do Departamento de Comércio dos Estados Unidos responsável pela promoção de competitividade e inovação industrial do país. O órgão desenvolve diversos documentos contendo as melhores práticas que podem ser adotadas livremente por qualquer setor do mercado norte americano. Para alguns departamentos do governo dos EUA, estas práticas devem ser obrigatoriamente seguidas.

Para a área de tecnologia e segurança da informação, o NIST desenvolveu uma série de metodologias (Série 800), incluindo uma específica para a realização de testes de segurança, denominada: “800-115 / Guia Técnico para Avaliações e Testes de Segurança da Informação”. Este guia está organizado em oito seções, sendo a Seção 1 introdutória, explicando o propósito e o escopo do documento. As demais seções estão divididas da seguinte forma:

- Seção 2 – Apresenta uma visão geral das avaliações de segurança da informação, incluindo políticas, papéis e responsabilidades, metodologias e técnicas;
- Seção 3 – Fornece uma descrição detalhada das várias técnicas de avaliação, incluindo análise de documentação, análise de logs, interceptação de dados na rede e verificação de integridade de arquivos;
- Seção 4 – Descreve várias técnicas para a identificação de alvos e procedimentos para mapear possíveis vulnerabilidades;
- Seção 5 – Explica as técnicas usadas para validar a existência de vulnerabilidades, tais como quebra de senha e testes de penetração;
- Seção 6 – Apresenta uma abordagem e processo de planejamento de uma avaliação de segurança;
- Seção 7 – Discute os fatores que são fundamentais para a execu-



OWASP Top 10 - 2013
The Ten Most Critical Web Application Security Risks

rele

+F Details About Risk Factors

Top 10 Risk Factor Summary

The following table presents a summary of the 2013 Top 10 Application Security Risks, and the risk factors we have assigned to each risk. These factors were determined based on the available statistics and the experience of the OWASP Top 10 team. To understand these risks for a particular application or organization, you must consider your own specific threat agents and business impacts. Even egregious software weaknesses may not present a serious risk if there are no threat agents in a position to perform the necessary attack or the business impact is negligible for the team involved.

RISK	Threat Agents	Attack Vectors	Prevalence	Discoverability	Technical Impact	Business Impact
A1 - Injection	App Specific	AVERT	COMMON	AVERAGE	SEVERE	App Specific
A2 - Broken Authentication and Session Management	App Specific	AVERAGE	MODERATE	AVERAGE	SEVERE	App Specific
A3 - Cross-Site Scripting (XSS)	XSS flaws without browser or other p	AVERAGE	VERY COMMON	LOW	MODERATE	App Specific
A4 - Insecure Direct Object References	A direct or indirect access to sensitive data	AVERAGE	COMMON	LOW	MODERATE	App Specific
A5 - Security Misconfiguration	Good security management should be in place	AVERT	COMMON	LOW	MODERATE	App Specific
A6 - Sensitive Data Exposure	Many web applications do not encrypt sensitive data	AVERT	COMMON	LOW	MODERATE	App Specific
A7 - Missing Function Level Access Control	Most web applications do not have proper access control	AVERT	COMMON	LOW	MODERATE	App Specific
A8 - Cross-Site Request Forgery (CSRF)	A CSRF attack can be used to perform actions on behalf of the user	AVERT	COMMON	LOW	MODERATE	App Specific
A9 - Using Components with Known Vulnerabilities	Components that have not been updated to the latest version can contain vulnerabilities	AVERT	COMMON	LOW	MODERATE	App Specific
A10 - Unvalidated Redirects and Forwards	Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.	AVERT	COMMON	LOW	MODERATE	App Specific

Additional Risks to Consider

The Top 10 cover a lot of ground, but there are many other risks you should consider and evaluate in your organization. Some of these have appeared in previous versions of the Top 10, and others have not, including new attack techniques that are being identified all the time. Other important application security risks (in alphabetical order) that you should also consider include:

- Clickjacking
- Content Security Policy (CSP)
- Denial of Service (DDoS) (See 2004 Top 10 - Entry 2004-A6)
- Excessive Resource Consumption (CVE-2013-0166)
- Information Leakage and Exposure (See 2007 Top 10 - Entry 2007-A9)
- Insufficient Authentication (CVE-2013-0166)
- Insufficient Logging and Accountability (Related to 2007 Top 10 - Entry 2007-A9)
- Lack of Transport Encryption and Integrity (See 2007 Top 10 - Entry 2007-A9)
- Malicious File Execution (See 2007 Top 10 - Entry 2007-A9)
- User Privacy

Figura 2. Relatório OWASP – TOP 10 – 2013

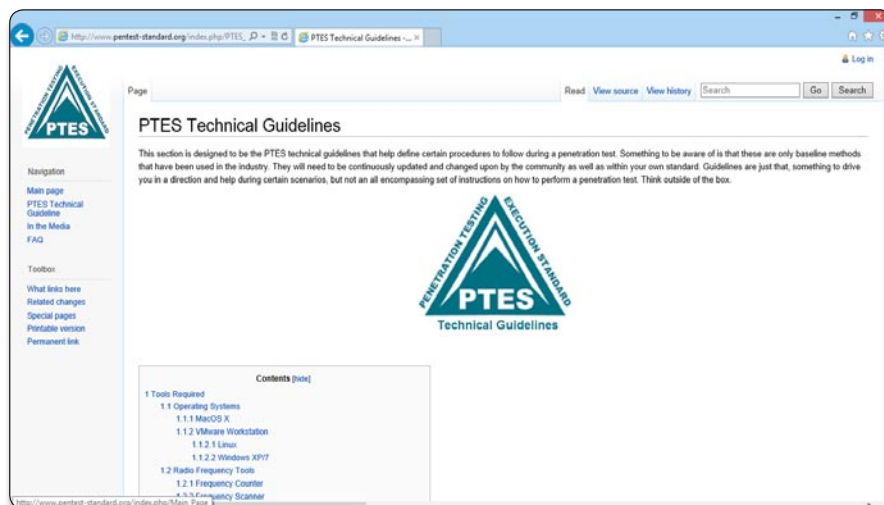


Figura 3. Site da entidade PTES contendo as diretrizes da norma

ção de avaliações de segurança, incluindo coordenação, a própria avaliação, análise e manipulação de dados;

- Seção 8 – Descreve uma abordagem para apresentação dos resultados da avaliação e fornece uma visão geral das ações para mitigar os riscos.

PTES – Penetration Testing Execution Standard

Uma das metodologias mais recentes e que vem ganhando espaço no mercado de Pentests é a PTES, que em uma tradução livre significa: “Padrão de Execução de Teste de Penetração” (Figura 3). Seu objetivo futuro é se tornar uma norma com diretrizes para padronizar a realização das avaliações técnicas de segurança por prestadores de serviços e dar suporte para que organizações que adquirem tal serviço possam exigir um trabalho dentro de um modelo de qualidade aceitável no mercado.

A metodologia PTES divide o processo de um Pentest em sete etapas:

- 1. Predefinição** – Antes de realizar a execução da avaliação técnica, o executor da tarefa, em conjunto com o cliente, deve definir o escopo do teste, metas, expectativas e custos. É nesta etapa que são assinados os termos de confidencialidade entre as partes (NDA – Non-Disclosure Agreement);
- 2. Coleta de Inteligência** – O executor do teste inicia o processo de coleta de informações sobre o alvo, incluindo funcionários, instalações, produtos, sistemas, etc. Tudo que possa ser relevante e que possa auxiliar na identificação de vulnerabilidades;
- 3. Modelagem de Ameaças** – Nesta etapa, o executor do teste analisa as informações coletadas para identificar quais as prováveis vulnerabilidades do alvo de acordo com o valor dos ativos da organização e o modelo de negócios do alvo;
- 4. Análise de Vulnerabilidade** – Nesta etapa são realizados testes utilizando ferramentas automatizadas em busca de informações sobre as vulnerabilidades e as melhores formas de explorá-las;
- 5. Exploração** – A etapa de exploração é o momento que o executor do Pentest obtém acesso a um sistema ou recursos, burlando controles de segurança que foram avaliados previamente;
- 6. Publicar Exploração** – Trata-se da documentação de todas as informações sensíveis do alvo, de forma que possa permitir a identificação de novos ativos a serem explorados, obtendo mais acesso à rede do alvo;
- 7. Relatório** – É a apresentação dos resultados do teste, incluindo as conclusões dos resultados alcançados, e o mais importante para o cliente, o direcionamento para eliminar ou reduzir os riscos de exploração destas vulnerabilidades.

Assim como a maioria das metodologias, o PTES também é uma entidade que depende da colaboração de profissionais, o que é muito importante, pois assim permite que especialistas de diferentes áreas possam expor suas percepções no processo de execução de um Pentest.

Analisando cada uma das metodologias, notamos que elas não obrigam o uso de uma ferramenta específica para a realização dos testes, embora algumas como a PTES incorporem uma lista de softwares que auxiliam o profissional na execução de suas atividades. A lista é extensa, porém, para facilitar o trabalho dos auditores, existem algumas distribuições Linux que agrupam as principais ferramentas de Pentest em um único sistema, como o Backtrack, Kali (atualização do Backtrack), Pentoo, Backbox, entre outras.

Para profissionais que queiram iniciar de forma correta o entendimento de todos os processos que envolvem um Pentest, estudar o PTES é um ótimo começo.

Os relatórios e manuais de todas estas metodologias podem ser baixados nos endereços das entidades na seção **Links**.

Quando deve ser feito?

Independente da metodologia adotada, a realização deste tipo de teste deve ser frequente, dentro de intervalos programados, ou sempre que ocorrerem mudanças drásticas no ambiente físico ou lógico da rede. Pode parecer exagero, mas não é, pois as constantes mudanças na área de tecnologia, com novos recursos, novas ferramentas e novas soluções apresentadas dentro de um intervalo de tempo cada vez menor, trazem consigo inúmeros riscos, que podem ir desde vulnerabilidades não detectadas ou não previstas nos sistemas, até a adaptação de uso e pouco conhecimento das pessoas na operação do novo recurso. Somado a isso, temos uma

Conhecimento faz diferença!

engenharia de software magazine

Gerenciamento de Configuração
Definição + Ferramentas

Evolução do Software
Definições, preocupações e custos

Automação de Testes
Cuidados a serem tomados na implantação

+ de 290 vídeos para assinantes

Faça já sua assinatura digital! | www.devmedia.com.br/es

Faça um upgrade em sua carreira

Em um mercado cada vez mais focado em qualidade, ter conhecimentos aprofundados sobre requisitos, metodologia, análises, testes, entre outros, pode ser a diferença entre conquistar ou não uma boa posição profissional. Sabendo disso a DevMedia lançou uma publicação totalmente especializada em Engenharia de Software. Todos os meses você pode encontrar artigos sobre Metodologias Ágeis; Metodologias tradicionais (document driven); ALM (application lifecycle); SOA (aplicações orientadas a serviços); Análise de sistemas; Modelagem; Métricas; Orientação à Objetos; UML; testes e muito mais. **Assine Já!**

constante evolução de ameaças provenientes de pessoas mal intencionadas que encontram nestas brechas oportunidades para obter algum tipo de vantagem, especialmente financeira.

Algumas metodologias informam em seus requisitos com que frequência devem ser realizados os testes. Como exemplo, temos as empresas ligadas ao setor de Cartões de Crédito, que coletam, processam, armazenam e/ou transmitem informações destes cartões, obrigadas a adotar a norma PCI-DSS, que exige da organização a realização de testes de penetração anualmente e varreduras de vulnerabilidades trimestrais ou quando há mudanças no ambiente de rede.

Onde deve ser realizado?

Como vimos anteriormente, algumas normas e metodologias especificam pontos a serem avaliados, como aplicações web, perímetros da rede como firewalls e dispositivos wireless, além de avaliações internas dos servidores, chegando até a avaliar pessoas e suas funções, especialmente no universo dos colaboradores internos da organização. De forma geral, o responsável pelo teste deve ter em mente que a avaliação não pode ser engessada, ou seja, ela deve atender as necessidades de cada cliente alvo. Para que isso aconteça, a definição do escopo deve ser acertada previamente junto com o cliente, verificando o cumprimento de normas e particularidades específicas de cada setor do mercado.

O que vai ser avaliado deve estar claramente definido entre o cliente e o prestador de serviço. O cliente pode precisar apenas da análise de uma aplicação Web específica, como pode também precisar de uma varredura em toda a rede, incluindo servidores, firewalls e dispositivos de conexão wireless. Como citamos no início do artigo, em alguns casos os Testes de Penetração envolvem até conceitos de Engenharia Social, fazendo com que os funcionários também passem pelo crivo da auditoria. Se o cliente não sabe ao certo onde será realizado o teste, quais ativos realmente precisam deste tipo de avaliação, ele terá em mãos orçamentos

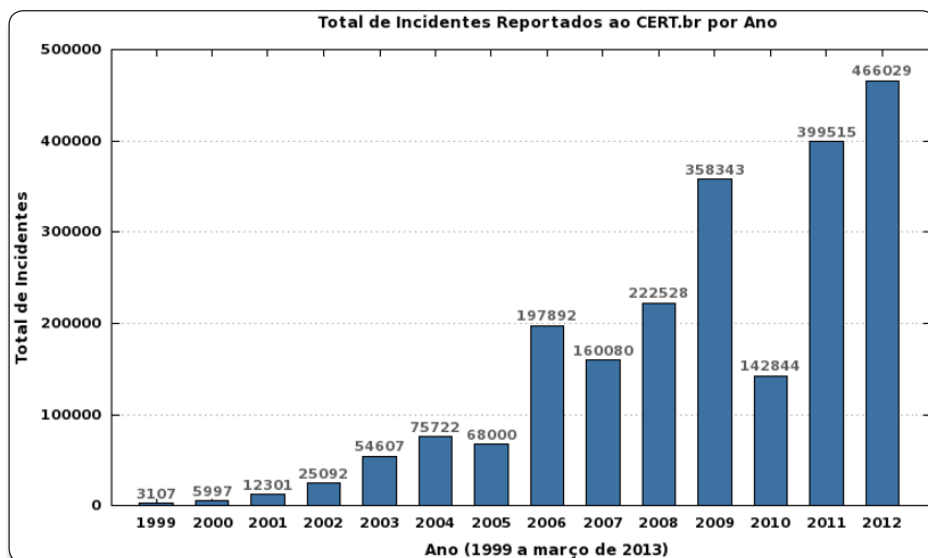


Figura 4. Histórico de incidentes reportados ao CERT.br

Incidentes Reportados ao CERT.br -- Janeiro a Dezembro de 2012



Figura 5. Tipos de ataque - CERT.br

de Pentests com valores completamente discrepantes, pois cada prestador irá indicar os alvos que em seu entendimento são mais importantes. Isso acontece justamente porque a parte contratante (o cliente alvo) não tem conhecimento básico do que é o Teste de Penetração e nem quais requisitos de segurança sua empresa precisa atender.

Por que tenho que fazer este teste?

Com base nas últimas estatísticas anuais de incidentes de segurança das redes conectadas à Internet brasileira, podemos notar índices recordes de ataques reportados ao “Centro de Estudos, Respostas e Tratamento de Incidentes de Segurança

no Brasil - CERT.br”. Somente em 2012, o CERT.br registrou um total de 466.029 alertas registrados (observe a Figura 4). É importante ressaltar que estes números servem apenas como uma estimativa, uma vez que muitas empresas não reportam os seus incidentes aos órgãos responsáveis.

Entre estes milhares de incidentes registrados, podemos notar que a maioria das notificações está relacionada à tentativa de exploração de vulnerabilidades na rede (Figura 5).

Para nossa melhor compreensão do que representa cada tipo de ataque notificado ao órgão, o CERT.br fornece em sua página as seguintes descrições:

- **Scan:** notificações de varreduras em redes de computadores, com o intuito de identificar quais computadores estão ativos e quais serviços estão sendo disponibilizados por eles. É amplamente utilizado por atacantes para identificar potenciais alvos, pois permite associar possíveis vulnerabilidades aos serviços habilitados em um computador;
- **Fraude:** segundo Houaiss, é “qualquer ato ardiloso, enganoso, de má-fé, com intuito de lesar ou ludibriar outrem, ou de não cumprir determinado dever; logro”. Esta categoria engloba as notificações de tentativas de fraudes, ou seja, de incidentes em que ocorre uma tentativa de obter vantagem;
- **Worm:** notificações de atividades maliciosas relacionadas com o processo automatizado de propagação de códigos maliciosos na rede;
- **Web:** um caso particular de ataque visando especificamente o comprometimento de servidores Web ou desfigurações de páginas na Internet;
- **Invasão:** um ataque bem sucedido que resulte no acesso não autorizado a um computador ou rede;
- **DoS (Denial of Service):** notificações de ataques de negação de serviço, onde o atacante utiliza um computador ou um conjunto de computadores para tirar de operação um serviço, computador ou rede;
- **Outros:** notificações de incidentes que não se enquadram nas categorias anteriores.

Conclusão

Em virtude dessa crescente tendência de ataques, cada vez mais as organizações, sejam elas públicas ou privadas, deverão rever suas políticas de segurança a fim de reduzir os riscos de ataques que possam comprometer suas operações. Consolidar as melhores práticas envolvendo ferramentas, processos e pessoas é a melhor forma de garantir um nível de segurança aceitável aos novos padrões de mercado.

Para atingir este objetivo, não podemos esperar que cada organização defina sua concepção do que é um Pentest. Unir todas as práticas em um conjunto de recomendações e diretrizes facilitará as atividades dos prestadores de serviço e poderá dar subsídios aos contratantes para validar a qualidade do trabalho realizado. Por falta de um padrão de avaliação, ainda encontramos dificuldades na execução deste tipo de trabalho, pois cada prestador procura oferecer um modelo de Pentest, adotando metodologias

próprias que muitas vezes podem deixar o contratante sem garantias de que estes investimentos em segurança realmente poderão lhe propiciar proteção efetiva e não uma falsa sensação de segurança.

Autor



Roberto Henrique

roberto@abctec.com.br

É Analista de Segurança da Informação na ABCtec, com 14 anos de experiência na área de TI (Suporte, Gestão, Consultoria), especializado em análise de vulnerabilidades e no tratamento de incidentes de segurança da informação. Formado em Análise e Desenvolvimento de

Sistemas e atualmente cursando a Pós-Graduação em Investigação de Fraudes e Forense Computacional, possui as certificações ISFS ISO/IEC 27002 Certified, F-Secure Certified Technical Professional – FSCTP, D-Link DBC, Microsoft MCP/MCDST. Escreve artigos para sites e revistas especializadas em Tecnologia e Segurança da Informação. Foi membro do Núcleo de TI do CIESP - São Bernardo do Campo e atualmente é membro do Comitê Técnico ABNT/CB21:CE27 sobre Segurança da Informação.



Links:

Site do NIST – National Institute of Standards and Technology

<http://www.nist.gov>

Site OWASP – Open Web Application Security Project

<https://www.owasp.org>

Site OSSTMM – Open Source Security Testing Methodology Manual

<http://www.isecom.org>

Site PTES – Penetration Testing Execution Standard

<http://www.pentest-standard.org>

Site PCI-DSS – Payment Card Industry Data Security Standard

<https://pt.pcisecuritystandards.org>

Norma ABNT NBR ISO/IEC 27002:2005

<http://www.abntcatalogo.com.br/norma.aspx?ID=1532>

CERT.br – Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil

<http://www.cert.br/>

Blog do autor

<http://abctec.blogspot.com>

Segurança de servidores Linux com ferramentas open source

Uma abordagem analítica e comparativa de ferramentas open source

A necessidade de redução de custos vem mudando o cenário das infraestruturas de rede nos parques tecnológicos. Neste contexto, a adoção cada vez mais comum de serviços em *cloud computing* traz benefícios consideráveis. Além desta opção, há também os *Virtual Private Servers*, servidores remotos e virtualizados que operam sem a necessidade de adoção e manutenção de hardwares, *datacenters* e links de acesso para servidores locais.

O ambiente onde os *Virtual Private Servers* operam – principalmente pelo fato de sua disponibilidade na Internet – são críticos no tocante à segurança. Essa necessidade faz com que a implementação de *softwares* de segurança seja imprescindível para a manutenção do sistema operacional.

Com base nisto, o objetivo deste trabalho é disponibilizar um cenário onde se possam analisar os principais recursos de algumas ferramentas *open source* para prover segurança em *Virtual Private Servers* Linux. O tópico seguinte pretende fornecer ao leitor informações que deixem clara a importância da segurança de informação em ambientes de *cloud computing*. Depois, é demonstrada uma comparação entre algumas ferramentas que são categorizadas de acordo com o seu objetivo. No tópico seguinte um estudo de caso é abordado, com a implementação de uma ferramenta de cada categoria. Por fim, é exposta a conclusão.

Principais Requisitos de Segurança

A segurança de sistemas é hoje um requisito essencial, uma vez que a informação é um dos ativos mais importantes de uma organização ou um dos bens de maior valor para um indivíduo. Do ponto de vista de Goodrich

Resumo DevMan

Porque este artigo é útil:

O artigo apresenta alguns conceitos e ferramentas que podem ser importantes para administradores que buscam melhorar a segurança dos seus servidores Linux. Assim, conheceremos um pouco sobre as soluções Fail2ban, Snort, Suricata, IPPL, SentryTools, CipherDyne PSAD e ClamAV. Por fim, é apresentado um estudo de caso.

e Tamassia (2013), a segurança da informação pode ser definida nos termos do acrônimo C.I.D. (Confidencialidade, Integridade e Disponibilidade), que podem ser caracterizados como seguem:

- **Confidencialidade:** É a propriedade de se evitar acesso não autorizado à informação;
- **Integridade:** Trata-se da propriedade de garantia que a informação não foi alvo de alteração não autorizada;
- **Disponibilidade:** É a propriedade de que a informação estará acessível e modificável para usuários autorizados.

O fato de *Virtual Private Servers* possuir, na maioria dos casos, um cenário inseguro – dada a sua disponibilidade na internet – faz com que exista a preocupação com a segurança destes tipos de sistemas. Walker (2013) elenca as principais ameaças existentes em um cenário compartilhável, como o de *Cloud Computing*, que segue os mesmos princípios dos cenários dos *Virtual Private Servers*. Entre elas estão às ameaças de “violação de dados”, “perda de dados”, “uso de aplicações inseguras” e “negação de serviço”, analisados a seguir.

Violação de dados

A violação de dados é um incidente que ameaça a integridade da informação. Para se compreender a importância dessa preocupação

e, principalmente, para justificar a presente pesquisa, nos remetemos à Madureira (2013) que, por sua vez, afirma que – no ano de 2012 – o custo total médio de um incidente de violação de dados no Brasil foi de R\$ 2,64 milhões. O autor ainda complementa afirmando que há registros de casos aonde o prejuízo chega a quase R\$ 10 milhões com tal incidente.

Perda de dados

A perda de dados é um incidente que ameaça a disponibilidade da informação. Nesse sentido, o uso de *Virtual Private Servers* em *Cloud Computing*, segundo estudo da Symantec (2012), vem crescendo em decorrência de uma alternativa para pequenas e médias empresas contra a perda de dados. A pesquisa realizada contou com a colaboração de 2053 organizações em 30 países e revela que 34% (um total aproximado de 698 organizações) adotam o uso de nuvens públicas.

Uso de Aplicações Inseguras

Quando falamos em *Virtual Private Servers Linux* estamos falando de um servidor virtual, executado em um ambiente de *Cloud Computing* com um sistema operacional que opera sobre o Kernel Linux. Isso nos remete ao princípio do *open source*, uma vez que um sistema operacional Linux tem como principal vantagem os softwares agregados a ele e disponibilizados como código aberto – em geral esta disponibilização está discriminada em sua GPL.

Softwares como Apache e MySQL, devido ao uso massivo na internet, são constantes alvos de ataques. Grupos de segurança de redes ou mesmo *crackers* dedicam seu tempo e conhecimento a descobrir vulnerabilidades dos softwares *open source* em sua versão estável. Isso cria um ciclo infundável que pode ser visualizado na Figura 1.

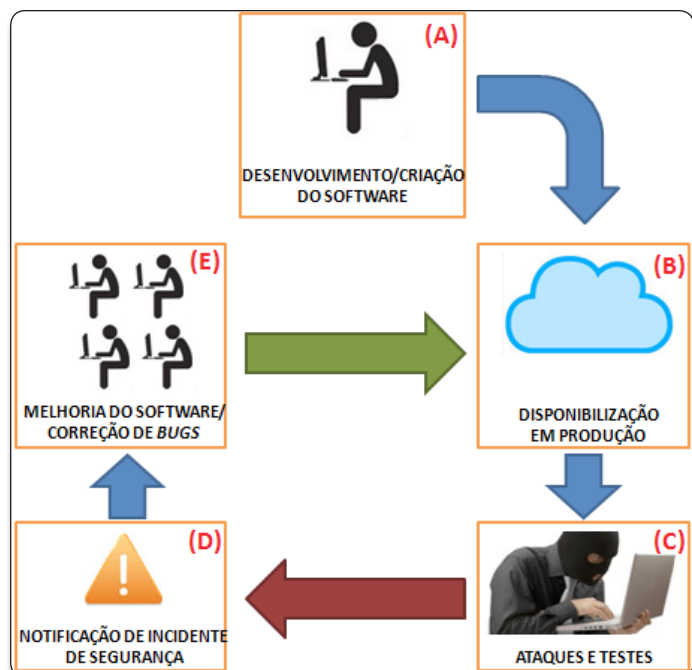


Figura 1. Ciclo de atualização de software motivada por incidente de segurança da informação

A Figura 1 pode ser interpretada da seguinte maneira, conforme a sua legenda alfabética:

a) Desenvolvimento/Criação do Software: Diz respeito ao momento em que o software está em desenvolvimento. Esse primeiro item está relacionado ao processo que vai desde o projeto de acordo com as necessidades até a finalização de um pacote estável;

b) Disponibilização em Produção: Diz respeito à disponibilização do software em versão estável (ou não) à comunidade. Pode ser resumido no fato da disponibilização do software para download na página do desenvolvedor;

c) Ataques e Testes: Um software – seja ele livre ou não – sempre que é disponibilizado e na medida em que a comunidade torna a sua aplicação popular, é alvo de ataques de *hackers* ou *crackers*. Os primeiros têm por objetivo descobrir vulnerabilidades e potenciais ameaças que o uso de tal software traz consigo. Já os segundos, além do mesmo objetivo dos *hackers*, querem explorar tais vulnerabilidades de forma a prejudicar a comunidade. Fato é que ambos contribuem – direta ou indiretamente – para o aprimoramento do software em questão;

d) Notificação de Incidente de Segurança: Tão logo uma vulnerabilidade é descoberta, a comunidade é comunicada por meio de listas de segurança de softwares. Um exemplo de lista é a “*Security Tracker*” (veja a seção **Links**), uma ferramenta que permite ao usuário receber notificações de incidentes de segurança relacionados a determinados softwares;

e) Melhoria do Software/Correção de Bugs: Última etapa do ciclo de atualização do software. A comunidade ou os desenvolvedores oficiais trabalham a fim de solucionar o bug reportado e devolver em produção do software atualizado, para que a etapa “a” seja executada novamente, fechando aqui um “ciclo infundável”.

Por mais controles que uma aplicação possua, ela nunca poderá ser considerada 100% segura. Isso é o que Santos (2008) afirma, baseando-se no princípio de que na medida em que a eficácia dos controles aumenta, os *crackers* aprimoram suas técnicas de invasão e exploração das vulnerabilidades.

Negação de Serviço

Ataques de negação de serviço são frequentes em redes de computadores conectadas à internet. Estes ataques são divididos em dois tipos, de acordo com a quantidade de máquinas atacantes. São eles:

- **DoS – Denial of Service:** Ataque de negação de serviço onde apenas um *host* envia um grande número de pacotes ao destino;
- **DDoS – Distributed Denial of Service:** Ataque de negação de serviço onde dois ou mais *hosts* enviam grande quantidade de pacotes ao destino.

Nos dois casos o objetivo é o mesmo: causar um cenário onde o servidor ocupa todo o seu processamento para responder às requisições dos atacantes, tornando demorada ou impossível a resposta de requisições de clientes reais.

Para se ter ideia da importância de se prevenir dessa ameaça, remetemo-nos a Anchises (2012). O autor afirma que no segundo semestre de 2011, o ataque de DDoS mais poderoso identificado pela Kaspersky foi 20% mais forte que o ataque mais forte realizado no semestre anterior. O autor complementa afirmando que o foco principal dos ataques tem como alvo páginas de comércio eletrônico e se usa de uma técnica denominada HTTP flood (alto número de envio de requisições HTTP, causando a sobrecarga do servidor web).

Comparativo de Ferramentas open source

Ferramentas *open source* que agregam segurança em *Virtual Private Server* Linux são uma importante alternativa para usuários ou empresas que não desejam (ou não possam) investir consideravelmente na compra de *softwares* proprietários ou aquisição de licenças. Em contrapartida, é necessário um preparo do administrador do sistema para a manutenção dessas ferramentas em produção. Algumas destas ferramentas serão estudadas no decorrer deste trabalho, com o objetivo de proporcionar ao leitor um cenário analítico entre elas, destacando suas vantagens e desvantagens. Elas se dividem nas seguintes categorias:

- IDS/IPS (*Intrusion Detection System/Intrusion Prevention System*);
- NSM (*Network Security Monitoring*);
- LAS (*Log Analysis System*);
- Antivírus.

A **Tabela 1** mostra de forma dividida os *softwares* que serão analisados, de acordo com a sua categoria.

IDS/IPS	NSM	LAS	Antivírus
- Fail2ban - Snort - Suricata	- IPPL - Sentry Tools	- CipherDyne PSAD	- ClamAV

Tabela 1. Categorização dos softwares de acordo com sua natureza

Intrusion Detection System e Intrusion Prevention System

Sistemas de detecção e/ou prevenção de intrusão são *softwares* que possuem uma base de dados de comportamentos suspeitos e analisam o tráfego na rede ou o comportamento de determinados *softwares* secundários a fim de identificar se se trata de uma tentativa de intrusão. Existem IDS e IPS baseados em *hardware* (HIDS e HIPS) e IDS e IPS baseados em *host* (HIDS e HIPS). Os IDS e IPS analisados neste tópico são baseados em *host*.

Fail2ban

De acordo com a página oficial do projeto Fail2ban (2013), o *software* trabalha analisando arquivos de *log* de conexões e “banindo” endereços IP com comportamentos maliciosos.

O Fail2ban pode ser definido como uma ferramenta que protege uma variedade de serviços contra tentativas de acessos indesejados: “[...] *fail2ban* is a tool that serves to protect a variety of services against unwanted visitors [...]” (HOBSON, 2013, p. 161)

Um exemplo é a tentativa de *login* em determinado serviço (vsFTPd, por exemplo) com usuário(s) e senha(s) inválido(s). Para cada tentativa de login frustrada, o serviço (nesse caso o servidor FTP) gera uma linha de texto nova no *log* (*vsftpd.log*), sendo um alerta para o Fail2ban, que está a monitorar esse *log*.

Detalhes da implementação e funcionamento deste *software* serão abordados no tópico “Estudo de Caso”.

Snort

A página oficial do projeto Snort (2013) o define como o sistema de detecção e prevenção de intrusões mais utilizado no mundo, com cerca de 400 mil usuários registrados.

Em uma definição ainda mais detalhada e técnica da ferramenta Snort, Cox e Gerg (2007) o caracterizam como o provável melhor *open source* IDS disponível. Os autores complementam a caracterização da ferramenta afirmando que ele foi inicialmente desenvolvido para operar via linha de comando (sem *interface* gráfica), mas que, devido à sua popularidade e desempenho, várias aplicações desenvolvidas por terceiros passaram a integrá-lo (inclusive algumas *interfaces* gráficas), o que o torna o melhor *open source* IDS disponível para *download*.

Caswell, Beale e Baker (2007) apresentam uma definição alternativa do Snort, afirmando ser uma ferramenta de detecção de intrusão *open source* capaz de realizar uma análise em tempo real do tráfego e registrar *logs* de pacotes em redes IP para posterior estudo.

O *software* Snort pode ser configurado para executar em três diferentes modos, de acordo com a página oficial do projeto (2013):

- 1. Sniffer mode** – Nesse modo o Snort apenas exibe na saída padrão – monitor de vídeo, por exemplo – os pacotes que estão sendo trafegados pela rede;
- 2. Packet logger mode** – Similar ao *sniffer mode*, porém armazena os dados coletados em um arquivo no disco rígido;
- 3. Network IDS** – É o modo que apresenta mais flexibilidade de configuração. Aqui ele opera como detector e analisador do tráfego, de forma a identificar tentativas de acessos não autorizados.

O Snort se mostra uma ferramenta mais robusta e com muitos recursos adicionais em comparação com o Fail2ban. Enquanto o segundo apenas trabalha buscando padrões de *strings* em arquivos de *log*, o Snort possui ferramentas que possibilitam uma análise mais inteligente do tráfego, visto seus três modos de operação.

Suricata

Suricata é um IDS, IPS e NSM *open source* que se destaca pelo seu alto desempenho. Ele foi desenvolvido e é mantido pela OISF (*Open Information Security Foundation* ou Fundação de desenvolvimento de softwares). Existem – de acordo com a mesma fonte – três motivos para ao menos se testar o Suricata. São eles:

1. É um sistema *multithreaded*, ou seja, suporta processamentos simultâneos no mesmo sistema operacional. Possui a flexibilidade de se configurar qual é o máximo de carga que cada instância de execução do Suricata vai poder exercer em cada núcleo de processamento;

2. Tem suporte à identificação de protocolo, de forma nativa. Esse recurso possibilita ao administrador da rede a criação de regras por protocolos e não por portas, o que de certa forma é o mais inteligente;

3. Nativamente o Suricata é capaz de identificar centenas de tipos de arquivos que estejam trafegando pela rede, pois possui uma base de dados que o possibilita tal ação. Essa *feature* pode ser útil na necessidade de se identificar determinada transferência na rede local, como por exemplo, a disseminação de um vírus.

O Suricata é uma excelente alternativa ao Fail2ban, porém não se deve comparar os seus recursos com a forma de operação do Snort, que é mais completo por possuir modos de operação diferentes em comparação com os outros dois IDS/IPS já citados.

Network Security Monitoring

Softwares de Network Security Monitoring têm o objetivo de analisar o comportamento da rede e – em caso de algum comportamento anormal – notificar o administrador ou realizar ações previamente configuradas.

IPPL

O IPPL (*Internet Protocol Packet Logger*) é um NSM, de acordo com a categorização exposta na **Tabela 1**. Hass e Bernard (2000), desenvolvedores do *software* definem o IPPL como um *daemon* (software independente que é executado em segundo plano) que registra em arquivos de *log* as informações dos pacotes IP destinados a determinado host na rede. Ele é executado em segundo plano e exibe informações sobre os pacotes IP. Além disso, possui um filtro onde se especifica o que monitorar e o que não monitorar. É um *software open source* licenciado sob GPL.

Em seus aspectos técnicos, os autores complementam detalhando o funcionamento interno do IPPL da seguinte maneira:

- **Log de Pacotes** – O IPPL, assim como o Suricata, é *multithreaded*. Para cada evento novo (cada novo protocolo registrado), a *thread* principal cria um *socket* e então as *subthreads* decidem, de acordo com os filtros, se ele será analisado ou não;
- **Mecanismo de Filtragem** – Para cada pacote recebido, uma lista de regras é lida e o seu conteúdo é comparado com o de cada item da lista. Caso o item da lista case com o tipo de pacote, ele é registrado;
- **Cache DNS** – Para tornar mais eficiente à resolução reversa de DNS, o IPPL cria uma base de dados de nomes, que é renovada de tempos em tempos. Caso esse recurso não fosse implementado pela equipe de desenvolvimento, cada pacote recebido necessitaria de uma consulta a um servidor DNS, causando a sua sobrecarga e também uma sobrecarga desnecessária na rede local.

SentryTools

A ferramenta, segundo afirma a página oficial do projeto (2013), provê segurança em nível de *host* implementando serviços para plataformas Unix. As ferramentas contidas no Sentry Tools são:

PortSentry, Logcheck/LogSentry e o HostSentry. Na sequência é apresentada uma breve descrição desses três softwares:

- **PortSentry**: Fonseca (2011) define o PortSentry como um sistema que faz uma espécie de “simulação”, abrindo portas no servidor – simulando serviços – e baseado no que receber de conexão nessas portas ele pode tomar alguma atitude, como por exemplo, solicitar ao *firewall* que bloqueie o endereço IP de origem;
- **Logcheck/LogSentry**: Sobre o LogSentry – antigamente chamado de Logcheck – Negus (2013) afirma ser uma ferramenta muito útil para que se possa gerenciar os arquivos de *log* do sistema. O *software* trabalha de forma a analisar os arquivos de *log* gerados pelo *syslog*, destacando as mensagens que possam caracterizar algum problema de segurança para o *host* ou o sistema operacional. (Syslog é um padrão criado pela IETF para transmissão de mensagens de log em redes IP);
- **HostSentry**: Rowland (2003) caracteriza o HostSentry como um sistema de detecção de intrusão baseado em *host*. Pode ser definido como um *Login Anomaly Detection*, pois monitora tentativas de *login* no *host* em busca de anomalias ou comportamentos suspeitos. Para isso, conta com uma base de dados que o auxilia na comparação em busca de ameaças.

Na comparação com o IPPL, o Sentry Tools tem a vantagem de ser modular. As três ferramentas descritas acima (PortSentry, LogSentry e HostSentry) trabalham independentes umas das outras, além de ter objetivos distintos na garantia de segurança de *Virtual Private Servers* Linux. O IPPL, como já citado, é apenas um *software* que registra as atividades de rede do *host*, tornando-o menos robusto, porém mais leve em comparação com o Sentry Tools.

Log Analysis System

Softwares de Log Analysis System têm como objetivo analisar arquivos de log gerados por outras aplicações em busca de padrões. Estes padrões podem indicar alguma tentativa de fraude na rede.

CipherDyne PSAD

O CipherDyne PSAD pode detectar vários tipos de tráfego suspeito na rede. RACH (2007).

A página oficial do projeto (ver seção **Links**) o define como uma coleção de *daemons* que – ao serem executados em *hosts* Linux – analisam os *logs* criados pelo *netfilter/iptables* em busca de comportamentos suspeitos na rede.

A **Figura 2** ilustra um exemplo de topologia de rede com acesso à internet onde o *software* pode ser usado. Nesse caso, o atacante – denominado *Scanner/Attacker* – sempre que necessitar acessar os dois ativos da rede local LAN – nomeados como *Web Server* e *DNS Server* – terá o tráfego analisado pelo *software*, denominado *iptables/psad*.

Antivírus

Softwares de Antivírus podem trabalhar de duas formas. A primeira é analisando o tráfego da rede em busca de padrões suspeitos de códigos maliciosos, porém no caso de conexões

criptografadas acaba não sendo eficaz. Já a segunda forma é analisando arquivos que estão armazenados em algum dispositivo como discos rígidos e *pen-drives*.

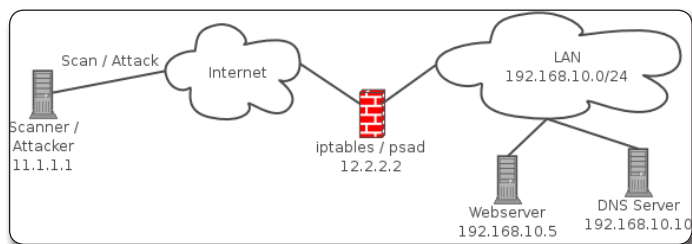


Figura 2. Estrutura típica de implementação do CipherDyne PSAD em uma rede LAN com acesso à Internet – Fonte: <http://cipherdyne.org/>

ClamAV

A página oficial do projeto (ver seção **Links**) define o ClamAV como um antivírus *open source* desenvolvido com o objetivo de detectar *Trojans*, Vírus, *Malwares*, dentre outras ameaças. Ele fornece *daemons* de alto desempenho (por ser *multithreaded*) além de ferramentas poderosas de linha de comando.

O ClamAV conta com uma biblioteca de padrões de vírus que é usada como base nas comparações dos arquivos para que o *software* seja capaz de determinar se determinado arquivo é suspeito ou não.

Turnbull (2005) afirma que uma vantagem significativa na escolha do ClamAV, além do fato de ser uma ferramenta *open source*, é a de possuir as definições (biblioteca) de vírus frequentemente atualizadas.

Estudo de Caso

Neste tópico, uma ferramenta de IDS/IPS (Fail2ban), uma ferramenta de NSM (IPPL) e uma ferramenta de LAS (Cipherdyne PSAD) elencadas na **Tabela 1** serão instaladas e testadas. O objetivo é analisar o processo de instalação, bem como algumas de suas funcionalidades básicas.

Instalação da Ferramenta Fail2ban e do servidor FTP vsFTPD

Para a instalação do IDS/IPS Fail2ban será utilizada a distribuição Debian na sua versão 6 com Kernel Linux 2.6.32. Neste processo será testada a eficácia do Fail2ban quando um cliente FTP tenta por diversas vezes se autenticar no servidor vsFTPD com credenciais inválidas, simulando uma tentativa de acesso não autorizado ou mesmo um ataque de negação de serviço.

Os *softwares* Fail2ban e vsFTPD foram instalados pelo gerenciador de pacotes padrão da distribuição. Para isto, execute os comandos:

```
root@localhost:~# apt-get update && apt-get -y install fail2ban
root@localhost:~# apt-get -y install vsftpd
```

Após a instalação com sucesso dos dois *softwares* e suas dependências, o arquivo `/etc/fail2ban/jail.conf` foi editado e a diretiva `[vsftpd]` alterada para o padrão apresentado na **Listagem 1**.

Listagem 1. Trecho de código do arquivo `/etc/fail2ban/jail.conf`.

```
[vsftpd]
enabled = true
port    = ftp,ftp-data,ftps,ftps-data
filter  = vsftpd
logpath = /var/log/vsftpd.log
maxretry = 6
```

As opções indicadas na listagem podem ser interpretadas da seguinte maneira:

- **enabled** – Pode ser **false** ou **true**. Essa opção habilita ou desabilita a função da diretiva `[vsftpd]`;
- **port** – Quais as portas que o Fail2ban deve bloquear assim que determinado comportamento suspeito for encontrado. As portas não são numéricas e as referências nominais estão em `/etc/services`;
- **filter** – Determina qual o tipo de filtro. O path de filtros é o diretório `/etc/fail2ban/filter.d/`;
- **logpath** – Qual o arquivo de log que o Fail2ban irá analisar em busca de comportamentos suspeitos. Padrões contidos no filtro especificado na opção **filter** serão buscados nesse arquivo;
- **maxretry** – Número de tentativas de login no servidor vsFTPD sem sucesso que o Fail2ban deve interpretar como normal. Números de tentativas maiores que o valor especificado nessa opção serão interpretados como ataque.

Após a realização destes ajustes, basta iniciar o serviço Fail2ban com a execução do script:

```
root@localhost:~# /etc/init.d/fail2ban start
```

O servidor FTP precisará apenas de dois ajustes na sua configuração contida em `/etc/vsftpd.conf`:

```
anonymous_enable=NO
local_enable=YES
```

A opção **anonymous_enable=NO**, somada à opção **local_enable=YES**, indica que apenas usuários do sistema poderão fazer login no servidor FTP. Para iniciá-lo, basta executar o script conforme o comando:

```
root@localhost:~# /etc/init.d/vsftpd start
```

Para verificar se o servidor FTP está em execução, executa-se o comando `ftp` e realiza-se o login com o usuário “anonymous” e senha nula. A informação esperada é “230 Login successful”, que pode ser vista na **Listagem 2**.

Simularemos agora uma tentativa de login com usuário e senha desconhecidos: “user” e “password”. O objetivo deste teste é analisar a saída de texto que o servidor vsFTPD adicionará no log `/var/log/vsftpd.log`. A saída do log é apresentada a seguir:

```
Fri Jul 12 12:20:13 2013 [pid 2] CONNECT: Client "127.0.0.1"  
Fri Jul 12 12:20:21 2013 [pid 1] [user] FAIL LOGIN: Client "127.0.0.1"
```

Com um cliente FTP, simularemos agora uma tentativa de *login* via rede com seis tentativas inválidas, conforme ilustra a **Figura 3**.

Listagem 2. Teste de acesso ao servidor FTP.

```
root@localhost:~# ftp localhost  
Connected to localhost.  
220 (vsFTPd 2.3.2)  
Name (localhost:root): anonymous  
331 Please specify the password.  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>
```



Figura 3. Tentativas de fraude ao servidor FTP vsFTPD executados pelo cliente FileZilla

Após a tentativa de fraude, o log do IDS/IPS Fail2ban mostra que o mesmo reconheceu a tentativa de fraude, como pode ser verificado a seguir:

```
root@localhost:~# tail -1 /var/log/fail2ban.log  
2013-07-12 12:31:38,050 fail2ban.actions: WARNING [vsftpd] Ban 192.168.56.1
```

O trecho “Ban 192.168.56.1” indica que o Fail2ban comunicou o firewall netfilter/iptables solicitando o bloqueio do endereço IP 192.168.56.1 às portas FTP do servidor, o que pode ser verificado com a saída do comando:

```
root@localhost:~# iptables-save | grep 192.168.56.1  
-A fail2ban-vsftpd -s 192.168.56.1/32 -j DROP
```

A saída deste comando mostra que existe uma chamada **DROP** para a origem **192.168.56.101**.

Para provar que o firewall realmente bloqueou o acesso, novamente foi testada a conexão FTP ao servidor, o que pode ser visto na **Figura 4**.

Neste momento conclui-se – em relação à ferramenta Fail2ban – que se trata de um *software* extremamente simples de ser instalado e configurado, além de funcional. O IDS/IPS agiu conforme o

esperado, assim que identificou uma tentativa de fraude no *login* ao servidor FTP, comunicou-se com o *firewall* e solicitou o bloqueio do endereço IP de origem.

```
Estado: Conectando 192.168.56.101:21...  
Erro: A conexão excedeu limite de tempo  
Erro: Não foi possível conectar ao servidor
```

Figura 4. Tentativa de login no servidor FTP após bloqueio realizado pelo Firewall

Instalação da ferramenta IPPL

A instalação da ferramenta IPPL será realizada no mesmo sistema operacional utilizado na instalação da ferramenta Fail2ban exposta no tópico anterior.

Por se tratar apenas de um *daemon* que armazenará em arquivo de *log* as conexões desejadas para posterior auditoria (se necessário), sua instalação e configuração são bastante simples e podem ser exemplificadas conforme analisaremos a seguir.

A instalação do *software* foi realizada via gerenciador de pacotes padrão da distribuição, da seguinte forma:

```
root@localhost:~# apt-get install ippl
```

Após a instalação, configuraremos o IPPL para gravar em *log* todas as conexões entrantes ao *host*. Deste modo, no arquivo */etc/ippl.conf*, basta adicionar a seguinte diretiva:

```
log-in all /var/log/ippl/all.log
```

Feito isso, inicie o software com o comando *ippl &* para que o mesmo seja executado em segundo plano. Observe que foram realizadas três tipos de conexões entrantes ao *host*, sendo elas: SSH, FTP e ICMP ECHO REQUEST. O arquivo de log */var/log/ippl/all.log* ficou com o conteúdo apresentado na **Listagem 3**.

Listagem 3. Conteúdo do arquivo de log all.log.

```
root@localhost:~# cat /var/log/ippl/all.log  
Jul 12 13:17:28 ssh connection attempt from 127.0.0.1  
Jul 12 13:17:32 ftp connection attempt from 192.168.56.1  
Jul 12 13:17:48 ICMP message type echo request from 192.168.56.1
```

Em relação à ferramenta IPPL, conclui-se que é um poderoso aliado do administrador do sistema em um momento de análise ou auditoria. Após a ocorrência de incidentes ou mesmo para se analisar o comportamento da rede, um *log* de ocorrências é muito útil. Por se tratar de um simples arquivo de texto, filtros podem ser usados para se buscar padrões ou expressões desejadas.

Instalação da ferramenta CIPHERDYNE PSAD

A instalação da ferramenta PSAD será realizada no mesmo sistema operacional dos softwares analisados nos dois tópicos anteriores. De modo semelhante, foi utilizado o gerenciador de pacotes padrão da distribuição para essa instalação:

```
root@localserver:~# apt-get install psad
```

Como o PSAD trabalha analisando *logs* de firewall, nada foi alterado em seu arquivo de configuração e o mesmo foi iniciado com o comando a seguir:

```
root@localserver:~# /etc/init.d/psad start
```

A ferramenta *iptables* para o firewall *netfilter* foi utilizada para habilitar a gravação de *logs* de conexões entrantes na interface de rede **eth1** em */var/log/messages*. Deste modo, execute o comando:

```
root@localserver:~# iptables -I INPUT -i eth1 -j LOG
```

Após isso, a ferramenta Zenmap NMAP GUI (um scanner de rede gráfico que executa o NMAP em background) foi utilizada para que um *scanner* fosse executado no servidor, conforme ilustra a **Figura 5**.

Tão logo o scanner foi executado, o CipherDyne PSAD já percebeu o comportamento na rede e criou em seu diretório de *logs* (*/var/log/psad*) o diretório 192.168.56.1, que é justamente o endereço IP de origem do atacante (ou *scanner*). O conteúdo do diretório pode ser verificado com a saída do comando exposto na **Listagem 4**.

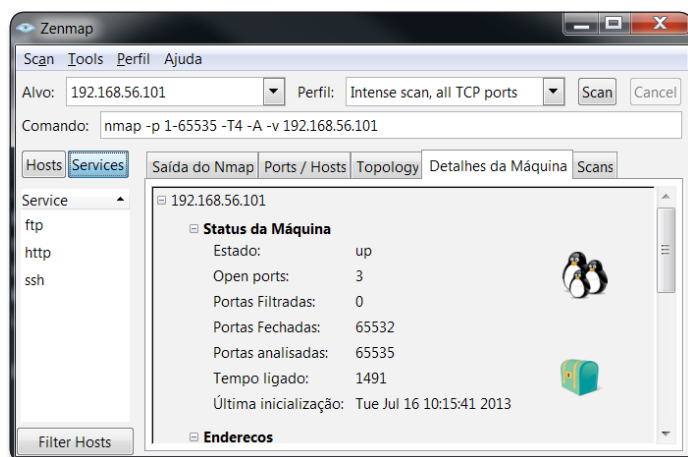


Figura 5. Execução de Scanner de rede no sistema operacional com o CipherDyne PSAD em execução

Listagem 4. Comando a ser executado e o seu resultado.

```
root@localserver:~# ls -l /var/log/psad/192.168.56.1/  
192.168.56.1_email_alert  
192.168.56.1_packet_ctr  
192.168.56.1_start_time  
danger_level
```

CURSOS ONLINE

A Revista Java Magazine oferece aos seus assinantes uma série de Cursos Online de alto padrão de qualidade.



CONHEÇA ALGUNS DOS CURSOS:

- Curso de noSQL (Redis) com Java
- Curso Básico de JDBC
- Java Básico: Aplicações Desktop
- JSF com Primefaces
- Conhecendo o Apache Struts

Para mais informações :
www.devmedia.com.br/curso/javamagazine
(21) 3382-5038



Cada um destes arquivos tem uma função importante na análise do administrador do sistema operacional. Esses arquivos são explicados a seguir:

- **192.168.56.101_email_alert** – Trata-se do conteúdo do e-mail que é enviado ao administrador do sistema – caso configurado e especificado em `/etc/psad/psad.conf`;
- **192.168.56.101_packet_ctr** – Representa um contador de pacotes. Cria-se no conteúdo desse arquivo uma divisão por protocolos (TCP ou UDP ou ICMP).
- **192.168.56.101_start_time** – Explicita a data e horário do início da execução do *scanner*. Especificado em *timestamp* no conteúdo do arquivo.
- **danger_level** – Define o nível de “perigo” que o *scanner* apresenta de acordo com as informações que conseguiu coletar. Vai de 0 a 5 e pode ser personalizado em `/etc/psad/psad.conf`.

Quanto à ferramenta PSAD, pode-se dizer que ela cumpre com o que promete. Sem necessidade de configuração inicial personalizada e apenas habilitando o *firewall netfilter* por meio do utilitário *iptables* o software foi capaz de identificar um *scanner* na rede local. O fato de a ferramenta notificar o administrador via e-mail também é um diferencial.

Conclusão

O estudo de caso proposto e executado mostrou que – de forma simples, gratuita e intuitiva – é possível agregar segurança em *Virtual Private Servers* Linux de uma forma que o administrador tenha autonomia para tomar decisões em relação à segurança do sistema. Outros sistemas operacionais possuem *softwares* de IDS/IPS, NSM, LAS e Antivírus similares, contudo este estudo de caso fortalecido pela revisão bibliográfica realizada mostrou a eficácia destes *softwares* de segurança em distribuições Linux, com o diferencial agregado do *software open source*.

No decorrer deste artigo alguns *softwares* foram analisados. Em resumo, quanto às ferramentas de IDS/IPS, o Snort foi a que apresentou mais recursos e uma forma mais modularizada de trabalhar, embora o Fail2ban seja a ferramenta de implementação e utilização mais simples dentre as demais de sua categoria. As ferramentas de NSM, por sua vez, embora apenas a IPPL tenha sido implementada, têm a vantagem de auxiliar e resguardar o administrador em casos de pós-ataque (uma perícia, por exemplo). Deste modo, é aconselhado que se adote soluções assim em conjunto com ferramentas de IDS e IPS para que a segurança da rede

não seja trocada apenas por um “monitoramento” ou “logging de eventos”. Quanto à ferramenta de LAS, CIPHERDYNE PSAD, pode-se dizer que é bastante útil, pois trabalha de forma reativa, analítica, e deve ser utilizada em conjunto com a função de *log* do *netfilter*.

Conclui-se, portanto, que, à medida que a necessidade de segurança se torna maior, *softwares* e tecnologias surgem com o objetivo de garantir a confidencialidade, a integridade e a disponibilidade dos dados servidos.

Autor

Thiago José Lucas

Professor Assistente na Fatec Ourinhos. Especialista em Projeto e Implementação de Redes de Computadores pela UTFPR e Tecnólogo em Segurança da Informação, também pela Fatec Ourinhos.



Autor

André Domingues

Professor Mestre da Universidade Tecnológica Federal do Paraná (UTFPR)



Links:

Security Tracker – site com lista de vulnerabilidades

<http://securitytracker.com>

Site da ferramenta PSAD

<http://cipherydyne.org/psad/>

Site da ferramenta ClamAV

<http://www.clamav.net/>

Site do Sentry Tools

<http://sentrytools.sourceforge.net/>

Site da ferramenta Snort

<http://www.snort.org/>

Site do Projeto IPPL

<http://pltp.net/ippl/>

Site do HostSentry

<http://www.securityfocus.com/tools/275>

Site do Fail2ban

http://www.fail2ban.org/wiki/index.php/Main_Page

Somos tão apaixonados por tecnologia que o nome da empresa diz tudo.

Porta 80 é o melhor que a Internet
pode oferecer para sua empresa.

Já completamos 8 anos e
estamos a caminho dos 80, junto
com nossos clientes.

Adoramos tecnologia.
Somos uma equipe composta
de gente que entende e
gosta do que faz,
assim como você.



Estrutura

100% NACIONAL.
Servidores de primeira
linha, links de alta
capacidade.



Suporte diferenciado

Treinamos nossa equipe
para fazer mais e melhor.
Muito além do esperado.



Serviços

Oferecemos a tecnologia
mais moderna, serviços
diferenciados e
atençados com as suas
necessidades.



1-to-1

Conhecemos nossos
clientes. Atendemos
cada necessidade de
forma única.
Conheça!



Porta 80
WEB HOSTING

Hospedagem | Cloud Computing | Dedicados | VoIP | Ecommerce |
Aplicações | Streaming | Email corporativo

porta80.com.br | comercial@porta80.com.br | twitter.com/porta80

SP 4063-8616 | RJ 4063-5092 | MG 4063-8120 | DF 4063-7486